



# BUSINESS & PRIVACY ISSUES IN DATA ANALYSIS

Data Anonymization Assignment

Supervisor Professor: Manolis Terrovitis



MAY 10, 2023

DIMITRIOS MATSANGANIS, f2822212  
FOTEINI NEFELI NOUSKALI, f2822213

MSc. Business Analytics FT 2022-2023



## Contents

Contents .....	2
Table of Figures.....	3
Description of the Assignment.....	5
Exercise A.....	7
Data Preprocessing .....	7
A1: The Quasi-Identifiers variables and the reasoning behind them .....	8
A2: Different Approaches to Data Protection: Anonymization, Pseudonymization, and Encryption under GDPR .....	12
A3: Understanding Identification: How Can a Person Be Identified .....	13
A4: Define differential privacy and explain the importance of the privacy parameter $\epsilon$ . .....	13
Exercise B.....	14
B1: Use the Amnesia anonymization tool to apply K-Anonymity to the dataset .....	14
Load the dataset .....	14
Generalization Hierarchies.....	17
Compare the Quasi-Identifiers Original vs Anonymized Dataset .....	30
Final Anonymized Dataset by Amnesia's K-Anonymity Generalization Hierarchies Methods.....	36
Differential Privacy.....	37
Differential Privacy - The Python library Diffprivlib.....	38
B2: Distribution of Numeric Features in the Dataset Using Histograms.....	39
B3: Applying Differential Privacy to Numeric Columns with Gaussian Mechanism .....	40
B4: Calculating Differentially Private Averages for Numeric Features in a Dataset .....	41
B5: Analyzing the Effect of Differential Privacy on Numeric Features Distribution.....	41
Deliverables Archive .....	44
References .....	45

## Table of Figures

Amnesia Figure 1: Load the data (1/2).....	15
Amnesia Figure 2: Load the data (2/2).....	15
Amnesia Figure 3: Preview the dataset after the successfully load. ....	15
Amnesia Figure 4: Anonymization check. ....	16
Amnesia Figure 5: 'Relationship', 'Sex', 'Age', and 'Quarter of Birth' combination anonymous check... 16	
Amnesia Figure 6: Housing/Group Quarters (GQ) Unit Serial Number Hierarchy. ....	17
Amnesia Figure 7: Hierarchy Text File for the Housing/Group Quarters (GQ) Unit Serial Number field. 18	
Amnesia Figure 8: Housing/Group Quarters (GQ) Unit Serial Number Amnesia Process (1/2).....	18
Amnesia Figure 9: Housing/Group Quarters (GQ) Unit Serial Number Amnesia Process (2/2).....	19
Amnesia Figure 10: Amnesia Algorithms set up based on loaded hierarchies (1/2). ....	19
Amnesia Figure 11: Amnesia Algorithms set up based on loaded hierarchies (2/2). ....	20
Amnesia Figure 12: Solution graph example with 3 variables. ....	20
Amnesia Figure 13: Solution graph example with 4 variables. ....	21
Amnesia Figure 14: Solution graph example with 5 variables. ....	21
Amnesia Figure 15: Anonymized Dataset preview prior to exportation (1/2). ....	22
Amnesia Figure 16: Anonymized Dataset preview prior to exportation (2/2). ....	22
Amnesia Figure 17: Quarter of Birth Hierarchy Text File. ....	23
Amnesia Figure 18: Race (Race Detailed Recode) Hierarchy Text File. ....	24
Amnesia Figure 19: Race (Race Detailed Recode) Hierarchy Loaded to Amnesia Graph. ....	24
Amnesia Figure 20: Hierarchy graph loaded into Amnesia, representing the Hispanic or Latino Origin variable with the Central American, Caribbean, South American, Spanish or Latino, and Not Hispanic or Latino categories.....	25
Amnesia Figure 21: Hierarchy text file to create the hierarchy graph into Amnesia for the Relationship (RELATE) field.....	26
Amnesia Figure 22: Hierarchy graph loaded into Amnesia, representing the Relationship variable with the Family_Member and Non-Family_Member categories. ....	26
Amnesia Figure 23: The Age hierarchy text with five categories based on 20-year intervals. ....	27
Amnesia Figure 24: Hierarchy graph loaded into Amnesia, representing the Age variable with five categories based on 20-year intervals. ....	27
Amnesia Figure 25: The Person Sequence Number (PNUM) hierarchy text with six subset categories. 28	
Amnesia Figure 26: Hierarchy graph loaded into Amnesia, representing the Person Sequence Number (PNUM) variable with six subset categories.....	28
Amnesia Figure 27: The Sex hierarchy text with two categories.....	29
Amnesia Figure 28: Hierarchy graph loaded into Amnesia, representing the Sex and the other Boolean variables with only two categories. ....	29
Amnesia Figure 29: Initial Dataset K-Anonymity Validation with 7 Quasi-Identifiers. ....	30
Amnesia Figure 30: : Anonymized Dataset K-Anonymity Validation with 7 Quasi-Identifiers.....	31
Amnesia Figure 31: Select the quasi-identifiers for the example from the Amnesia's dialog. ....	32
Amnesia Figure 32: Initial dataset k-anonymity validation with 3 quasi-identifiers, showing the suppression rate required to achieve a k-anonymity level of 3. ....	32

Amnesia Figure 33: Anonymized dataset k-anonymity validation with 3 quasi-identifiers, showing the significantly reduced suppression rate required to achieve the same k-anonymity level of 3 after implementing hierarchical generalization techniques. ....	33
Amnesia Figure 34: : Initial dataset k-anonymity validation with 5 quasi-identifiers, showing the suppression rate required to achieve a k-anonymity level of 3. ....	34
Amnesia Figure 35: Anonymized dataset k-anonymity validation with 5 quasi-identifiers, showing the significantly reduced suppression rate required to achieve the same k-anonymity level of 3 after implementing hierarchical generalization techniques. ....	34
Final Anonymized Dataset 1.....	36
Final Anonymized Dataset 2. ....	36
Final Anonymized Dataset 3. ....	37
Jupyter Notebook Figure 1: Age normalized frequency histogram.....	39
Jupyter Notebook Figure 2: Sequence Person number normalized frequency histogram. ....	40
Jupyter Notebook Figure 3: Age normalized frequency histogram after the added noise is implemented to the feature.....	42
Jupyter Notebook Figure 4: Sequence Personal Number normalized frequency histogram after the added noise is implemented to the feature.....	42
Jupyter Notebook Figure 6: Histogram errors results for different epsilon parameters values. ....	43

## Description of the Assignment

### Instructions

The US Census Bureau provides microdata to be used in research and applications. These data were deemed a threat to use privacy and in 2020 they were anonymized for the first time. In this exercise you must study the last non-anonymized census and discuss the privacy threats.

An example of the microdata of the 2010 census can be found here:

<https://www.census.gov/data/datasets/2010/dec/stateside-pums.html>

For a full description of the dataset you can read

<https://www2.census.gov/programs-surveys/decennial/2010/technicaldocumentation/complete-tech-docs/us-pums/pumsus.pdf>

For the exercise, please use the Delaware data:

[https://www2.census.gov/census\\_2010/12-Sideside\\_PUMS/Delaware/](https://www2.census.gov/census_2010/12-Sideside_PUMS/Delaware/)

### How to use the dataset

In order to use the dataset, download and decompress the zip file.

The extracted directory contains the **de.2010.pums.01.txt**

The de.2010.pums.01.txt file contains rows corresponding to:

- (a) Person records rows start with 'P'
- (b) Household records rows that start with 'H'

For the next steps ignore the Household records.

For the Person records please read the brief description for the comprised columns, and the possible values for each column that can be found in:

[https://www2.census.gov/census\\_2010/12-Sideside\\_PUMS/2010%20PUMS%20Record%20Layout.xlsx](https://www2.census.gov/census_2010/12-Sideside_PUMS/2010%20PUMS%20Record%20Layout.xlsx)

We suggest using software like Excel or Numbers (manually) to guide the Person data or write a script to parse the file into a table.

### Exercise A

After examining the data table (columns & values), answer the following questions:

1. Which attributes can act as quasi-identifiers and why?
2. Which of the following properties holds for the data?
  - a. They are anonymized
  - b. They are pseudonymized
  - c. They are encrypted

Explain the key differences between the three approaches with respect to GDPR.

3. Explain how a person can be identified.
4. Define differential privacy and explain the importance of the privacy parameter  $\epsilon$ .



## Exercise B

Load the dataset into a Python notebook (we suggest Jupyter) and display the first few rows to understand the data.

1. Use the Amnesia anonymization tool to apply k-anonymity to the dataset. Comment on the resulting dataset.
2. Plot the distribution of numeric features in the dataset using histograms.
3. Apply a random noise mechanism to some of the numeric columns using the Gaussian mechanism. The noise should be added to the original values in a way that preserves differential privacy.
4. Calculate the differentially private averages for the individuals using the noisy data.
5. Plot the distribution of numeric features after the noise addition. Try different values of the  $\epsilon$  parameter. Comment on the effect of the differential privacy on the results.

\* For steps 2-5 you can use the <https://github.com/IBM/differential-privacy-library> and <https://github.com/IBM/differential-privacy-library/tree/main/notebooks>.

## Exercise A

In the field of data privacy, the protection of personal information has become a significant concern. With the advent of new technologies, the amount of data collected and stored has grown exponentially. In this context, the General Data Protection Regulation (GDPR) was introduced to regulate the collection, processing, and storage of personal data within the European Union. One key aspect of GDPR is the protection of personal data through various methods such as anonymization, pseudonymization, and encryption. This exercise aims to explore the key features of these approaches and their implications for data privacy.

To be more precise, the following sections will examine a given data table and answer questions related to the attributes that can act as quasi-identifiers, the anonymization status of the data, and the key differences between anonymization, pseudonymization, and encryption with respect to GDPR. The sections will also discuss the concept of personal identification, the definition of differential privacy, and the importance of the privacy parameter  $\epsilon$  in ensuring data privacy. The aim is to provide a comprehensive understanding of data privacy and the different techniques used to protect personal data.

### Data Preprocessing

Preprocessing is an essential step in any data analysis project. It involves cleaning, transforming, and organizing raw data to make it suitable for further analysis. In many cases, data collected from various sources may have errors, missing values, or inconsistencies that can adversely affect the accuracy of the results. Therefore, preprocessing is necessary to remove these anomalies and improve the quality of the data.

In our case, preprocessing through R can be particularly beneficial. R is a powerful programming language used for statistical analysis, data visualization, and machine learning. It has numerous libraries and packages that can be used for preprocessing tasks, such as data cleaning, imputation, normalization, and scaling.

By using R for preprocessing, we can automate many of these tasks and ensure that the data is processed consistently and accurately. This can save time and effort, especially if we have large datasets that require manual cleaning and processing.

Overall, the ability to extract valuable insights from data is crucial in today's world. However, before data analysis can occur, data preprocessing must take place. Preprocessing refers to the cleaning, transformation, and reduction of raw data to make it more suitable for analysis. This report will focus on the preprocessing process of a dataset using the R programming language. The dataset being used is the 2010 Public Use Microdata Sample (PUMS). The purpose of the preprocessing is to anonymize the data and remove any identifying information while still retaining useful data for analysis.

It is very important to describe the procedure followed to the data preprocessing stage through R and the `pubs.R` script. The first step in the preprocessing process is to load the dataset into R. This is accomplished using the `readLines()` function, which reads the text file containing the data into R. The next step is to extract the words starting with "P" from the dataset. This is done using the `grep()` function with the `value` parameter set to `TRUE` to return the actual words instead of their index positions.

The extracted words are then split into substrings based on their length index. The length index specifies the number of characters that each substring should contain. The substrings are stored in a list using the `lapply()` function. The substrings are then combined into a dataframe using the `data.frame()` function. The resulting dataframe contains all of the relevant data from the original dataset, but without any identifying information.

The columns of the resulting dataframe are named according to the study that produced the original dataset. The columns that are not needed for analysis are then removed from the R environment for optimization and clarity purposes. The column names of the dataframe are then changed to more representative names using the `colnames()` function.

Finally, the Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, and Age columns are converted to integers using the `as.integer()` function. The resulting dataframes are then exported to CSV files for better visualization purposes and to enable their use in Python, the desired language for Exercise B. Additionally, the resulting dataframe is exported to a text file using the `write.table()` function.

In conclusion, the preprocessing process using R for the 2010 Public Use Microdata Sample (PUMS) dataset is a crucial step in preparing the data for analysis. The process involved loading the dataset, extracting relevant data, creating a dataframe, cleaning the dataframe, renaming the columns, converting data types, and exporting the resulting dataframes. The resulting dataframes are anonymized and contain only relevant data, allowing for safe and useful analysis.

### ***A1: The Quasi-Identifiers variables and the reasoning behind them***

Quasi-identifiers are attributes that, when combined, can potentially identify an individual or disclose sensitive information about them. In other words, they are attributes that can be used to link the data to the identity of an individual or a group of individuals. They are called quasi-identifiers because they are not explicitly identifiers like names or social security numbers, but they can be used in combination with each other and other data sources to re-identify individuals.

These attributes are considered quasi-identifiers because they could potentially be used in combination to re-identify individuals or disclose sensitive information about them. For example, some crucial identifying variable group combinations are the following three quasi-identifiers groups:

#### **Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, and Substituted Person Flag**

These three columns together could potentially be used to re-identify individuals in group quarters such as nursing homes or correctional facilities. The GQ Unit Serial Number identifies the specific group quarters unit, while the Person Sequence Number identifies the individual within that unit. The Substituted Person Flag indicates whether the individual is a substitute or not.

#### **Sex, Age, and Race Detailed Recode**

These three columns together could potentially be used to re-identify individuals based on their demographic characteristics. While no single column is a unique identifier, the combination of sex, age, and detailed race information could narrow down the possibilities and make it easier to re-identify individuals, especially if the population is small or the sample size is limited.

#### **Quarter of Birth, Hispanic or Latino Origin, and Same Sex Spouse Flag**

These three columns together could potentially be used to re-identify individuals based on their unique characteristics. While no single column is a unique identifier, the combination of these characteristics could make it easier to identify specific individuals. Specifically, using these three characteristics and taking into consideration the smaller population of the Hispanic or Latino origin individuals appearing in the particular dataset or concept the possibilities are shrined to a few individuals who match these characteristics. Along with other information the combined with these three characteristics the individuals can be re-identified with higher probabilistic accuracy.

To be more precise, we can provide an explanation for each attribute in the dataset to explain why it is or it is not considered a quasi-identifier:

- **Record Type (RECTYPE):** This field specifies whether the record pertains to a housing unit or a group quarters unit. While this field alone may not be a quasi-identifier in this case where only one kind of record type (e.g. Persons) is taking into consideration in an analysis framework.
- **Housing/Group Quarters (GQ) Unit Serial Number:** This field provides a unique identifier for the housing unit or group quarters unit. It is considered a quasi-identifier because it can be used in combination with Person Sequence Number, Relationship, Own Child Indicator or Related Child Indicator to identify individuals.



- **Person Sequence Number:** This field provides a unique identifier for each individual within a housing or group quarters unit. It is considered a quasi-identifier because it can be used in combination with Age, Sex, Race (e.g. Hispanic or Latino) to identify individuals.
- **Substituted Person Flag:** This field indicates whether the individual was substituted on the Census form by another member of the household. It is considered a quasi-identifier because it can be used in combination with Relationship, Group Quarters Type, Race to identify individuals.
- **Person weight:** This field indicates the weight assigned to the individual in the Census data. It is considered a quasi-identifier because it can be used in combination with Age, Sex, Race to identify individuals.
- **Relationship:** This field indicates the relationship of the individual to the head of the household. It is considered a quasi-identifier because it can be used in combination Age, Sex, Race to identify individuals.
- **Relationship Allocation Flag:** This field indicates whether the relationship was imputed (i.e., inferred) by the Census Bureau. It is considered a quasi-identifier because it can be used in combination with Age, Sex, Race to identify individuals.
- **Own Child Indicator:** This field indicates whether the individual is the biological child of the head of the household. It is considered a quasi-identifier because it can be used in combination with Age, Sex, Race to identify individuals.
- **Related Child Indicator:** This field indicates whether the individual is related to the head of the household as a child. It is considered a quasi-identifier because it can be used in combination with Age, Sex, Race to identify individuals.
- **Sex:** This field indicates the sex of the individual. It is considered a quasi-identifier because it can be used in combination with Age, Race, Group Quarters (GQ) Unit Serial Number to identify individuals.
- **Sex Allocation Flag:** This field indicates whether the sex was imputed (i.e., inferred) by the Census Bureau. It is considered a quasi-identifier because it can be used in combination with Age, Race, House District to identify individuals.
- **Same Sex Spouse Flag:** This field indicates whether the individual is in a same-sex marriage. It is considered a quasi-identifier because it can be used in combination with Quarter of Birth, Hispanic or Latino Origin, Age, Relationship, Relationship Allocation Flag to identify individuals.
- **Age:** This field indicates the age of the individual. It is considered a quasi-identifier because it can be used in combination with Sex, Race to identify individuals.
- **Age Allocation Flag:** This field indicates whether the age was imputed (i.e., inferred) by the Census Bureau. It is considered a quasi-identifier because it can be used in combination with Age, Quarter of Birth, Hispanic or Latino Origin, Same Sex Spouse Flag to identify individuals.
- **Quarter of Birth:** This attribute indicates the quarter of birth of a person. This can be used to identify the age of the person more accurately and hence can be considered a quasi-identifier in combination with Age, Hispanic or Latino Origin, Same Sex Spouse Flag.
- **Hispanic or Latino Origin:** This attribute indicates whether a person identifies as Hispanic or Latino or not. It is a quasi-identifier because it can be used to infer the ethnic or cultural background of a person. It can be considered a quasi-identifier in combination with Age, Quarter of Birth, Hispanic or Latino Origin, Same Sex Spouse Flag.
- **Hispanic or Latino Origin Allocation Flag:** This attribute is a flag indicating if the Hispanic or Latino Origin is allocated based on race. It is a quasi-identifier because it can be used in combination with the Hispanic or Latino Origin attribute to infer more information about a person's ethnic or cultural background. It can be considered a quasi-identifier in combination with Age, Quarter of Birth, Hispanic or Latino Origin, Same Sex Spouse Flag.
- **Number of Major Race Groups Marked:** This attribute indicates the number of major race groups a person belongs to. It is a quasi-identifier because it can be used to infer the race of a

person. It can be considered a quasi-identifier in combination with Race Detailed Recode, Race Short Recode, Race Checkbox Recode, White recode, Black or African American recode, American Indian and Alaska Native recode, Asian recode, Native Hawaiian recode, Other Pacific Islander recode, some other race recode.

- **White recode:** This attribute indicates whether a person identifies as White or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Age, Quarter of birth, Sex, Relationship, Own child indicator, Hispanic or Latino origin, Same sex spouse flag, Group quarters type, Housing/group quarters unit serial number.
- **Black or African American recode:** This attribute indicates whether a person identifies as Black or African American or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Relationship Allocation Flag, Own Child Indicator.
- **American Indian and Alaska Native recode:** This attribute indicates whether a person identifies as American Indian or Alaska Native or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with
- **Asian recode:** This attribute indicates whether a person identifies as Asian or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Relationship Allocation Flag, Own Child Indicator.
- **Native Hawaiian recode:** This attribute indicates whether a person identifies as Native Hawaiian or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Relationship Allocation Flag, Own Child Indicator.
- **Other Pacific Islander recode:** This attribute indicates whether a person identifies as Other Pacific Islander or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Relationship Allocation Flag, Own Child Indicator.
- **Some other race recode:** This attribute indicates whether a person identifies as Some Other Race or not. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Relationship Allocation Flag, Own Child Indicator.
- **Race Short Recode:** This attribute provides a short code for the person's race. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Age, Sex, Same Sex Spouse Flag, Quarter of Birth.
- **Race Detailed Recode:** This attribute provides a detailed code for the person's race. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Age, Sex, Same Sex Spouse Flag, Quarter of Birth.
- **Race Checkbox Recode:** This attribute provides a checkbox code for the person's race. It is a quasi-identifier because it can be used to infer the race of a person. It can be considered a quasi-identifier in combination with Age, Sex, Race, House unit.
- **Race Allocation Flag:** This attribute is a flag indicating whether race is allocated based on information from other attributes. It is a quasi-identifier because it can be used in combination with other race-related attributes to infer more information about a person's race. It can be considered a quasi-identifier in combination with Number of Major Race Groups Marked, White recode, Race, Race Checkbox Recode

- **Group Quarters Type:** This attribute indicates the type of group quarters where the person resides. It is a quasi-identifier because it can be used to infer the living situation of a person. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Own Child Indicator, Related Child Indicator, Sex.
- **Group Quarters Allocation Flag:** This attribute is a flag indicating whether the group quarters type is allocated based on information from other attributes. It is a quasi-identifier because it can be used in combination with the group quarters type attribute to infer more information about a person's living situation. It can be considered a quasi-identifier in combination with Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, Substituted Person Flag, Person weight, Relationship, Own Child Indicator, Related Child Indicator, Sex.
- **Padding:** This attribute refers to additional blank spaces added to the end of a record to make it a fixed length. It is not a quasi-identifier.

Conclusively, after the completion of an extensive analysis of the quasi-identifiers probable subgroups some dominant variables appear that play a significant role in to many quasit groups and they immerge as key factors that add the necessary information for the re-identification process.

In the given dataset, the following attributes can be regarded as main quasi-identifiers contributors variables:

- Age
- Sex
- House unit
- Persons Sequence Number
- Quarter of Birth
- Some races recode (Race Detailed Recode)
- Relationship

The use of the combination of Housing/Group Quarters Unit Serial Number, Person Sequence Number, Quarter of Birth, Relationship, Race, Age and Sex as quasi-identifiers poses a significant risk of re-identification of individuals in the dataset. This is because this group of quasi-identifiers is highly unique and sensitive, containing crucial personal information that can potentially lead to inductive identification. External datasets or knowledge sources that contain the same combination of quasi-identifiers along with additional information, such as full name or address, can be matched with the quasi-identifiers in the dataset, thereby enabling re-identification. The risk of re-identification is dependent on the uniqueness of the quasi-identifiers and the availability of external information that can be matched with them. To minimize the risk of re-identification, it is recommended to use a combination of quasi-identifiers that are less unique or sensitive. Additionally, proper de-identification techniques should be applied to ensure the anonymity and confidentiality of individuals in the dataset.

## ***A2: Different Approaches to Data Protection: Anonymization, Pseudonymization, and Encryption under GDPR***

The presented data is in a fully encoded form, where the fundamental identification elements are already removed. Specifically, the data is structured in a tabular format where each row entry represents a unique entity feature corresponding to a single person. At this stage, the data set can be **characterized as a pseudo-anonymized data set**. To further protect the data, various features of the data set are encoded. For example, the ethnicity feature is not presented as a string value, which directly exposes the ethnicity of the individual. Instead, it is assigned a code number that is linked to an ethnicity label. Additionally, the column names are removed from the stored data set, to prevent anyone from immediately knowing the features displayed for everyone in the data set. All these operations performed on the initial data set make it pseudo-anonymized and provide a first level of privacy protection, but in a reversible manner.

The dataset does not present any signs of data anonymization or encryption in its initial form. In order to explain explicitly the reasons of the data protection level classification we will cite the characteristics of each different protection method along with the main goal of each method and the legislative perception that has been adopted in the GDPR rule against each one of these three methods. The data anonymization method is a process of removing or modifying personal identifiable information in data sets to prevent the identification of individuals. The main goal of data anonymization is to make it impossible or extremely difficult to re-identify an individual. Data anonymization techniques include techniques such as generalization, randomization, suppression, and perturbation. Anonymization techniques implementation is irreversible and so it is impossible to reach and reattain the initial data form and granularity information. For this reason anonymized data it is not necessary to attain the data subject's consent in order to grant the data to various recipients of open data platforms. The key difference between anonymized data and other forms of data protection is that anonymized data does not fall under the scope of the GDPR as it is considered non-personal data anymore after they irreversible implementation of the anonymization techniques.

The data pseudo anonymization method is a process of replacing personal identifiable information with a pseudonym, or a code that does not directly identify an individual. Pseudonymization techniques include techniques such as hashing, tokenization, and encryption. Unlike data anonymization, pseudonymization does not completely remove personal identifiable information from data sets, but rather replaces it with a code or pseudonym. The implementation of pseudonymization techniques is completely reversible and the initial data can be reattained in their exact form. For this reason, the data that is only pseudonymized should be delivered only to trustworthy recipients that will process them mainly for scientific research purposes with the consent of the data subject and never offered or exposed as open data that can be accessed by several unknown parties. Pseudonymized data falls under the scope of the GDPR, but the regulation allows the processing of pseudonymized data for certain purposes such as scientific research and statistics.

The encryption method is a process of transforming data into a secret code to prevent unauthorized access. Encrypted data are encoded to be prevented from being read by unauthorized parties. Encryption is used mainly to ensure confidentiality, as a technical measure to protect data. The party doing the encryption uses an encryption key (collection of unique algorithms) to encode the data so that they appear unintelligible. The data can be re-read again, with the use of the encryption key from the recipient, but without it, the data cannot be accessed. Encryption techniques include symmetric encryption and asymmetric encryption. The key difference between encryption and other forms of data protection is that encryption does not modify or remove personal identifiable information, but rather protects it from unauthorized access. The encryption main purpose is to prevent the leakage of personal data to untrustworthy third parties, but it does not protect the data from the final recipient in case of a possibility that malicious practices are implemented by him. Encrypted data falls under the scope of the GDPR, and the regulation requires the implementation of appropriate technical and organizational measures to protect personal data.

### ***A3: Understanding Identification: How Can a Person Be Identified***

A possible re-identification attack could be carried out using the Housing/Group Quarters Unit Serial Number, Person Sequence Number, Quarter of Birth, Relationship, and Sex variables as quasi-identifiers. The attacker obtains a publicly available dataset that contains the quasi-identifiers mentioned above, as well as other non-sensitive variables. Firstly, the attacker selects a target person that they wish to re-identify in the dataset based on some external knowledge they possess, such as the target's name or address, then uses the external knowledge to narrow down the possible matches in the dataset to a small group of individuals. The quasi-identifiers (Housing/Group Quarters Unit Serial Number, Person Sequence Number, Quarter of Birth, Relationship, and Sex) are used to further refine the possible matches to the target person. Finally, the attacker checks the remaining individuals' attributes to determine which one is the target person and if the target person is successfully re-identified, the attacker can then use the dataset's other variables to gather sensitive information about the target person.

### ***A4: Define differential privacy and explain the importance of the privacy parameter $\epsilon$ .***

Differential privacy is a method of protecting the privacy of individuals when analyzing and sharing data. It provides a way to mathematically quantify the amount of information that is leaked when data is analyzed, and allows for the release of aggregate statistics without revealing individual-level information. The basic idea behind differential privacy is to add noise to the data before it is released, in such a way that the noise is indistinguishable from random noise. This makes it difficult for an attacker to determine whether a particular individual's data was included in the released data set. In the context of data privacy, random noise refers to the addition of random values to a dataset in order to mask or obscure sensitive information. This technique is commonly used in differential privacy to protect individuals' privacy while still allowing for accurate statistical analysis of the data. Random noise can be added in different ways, depending on the specific application and the type of data being protected. For example, in the case of numerical data, random noise can be added to the original values in order to mask their exact values. This can be done by adding a random value drawn from a pre-defined distribution to each value in the dataset. Alternatively, random noise can be added to categorical data by randomly changing some of the categories to other categories, while preserving the overall distribution of categories in the dataset. The amount of random noise that is added to a dataset can be controlled by a parameter called the "privacy budget." This parameter determines how much random noise can be added to the data without significantly impacting the accuracy of the statistical analysis.

The privacy parameter  $\epsilon$  is a crucial component of the differential privacy method. It determines the level of privacy protection provided by the method. In general, a smaller value of  $\epsilon$  provides a higher level of privacy protection, but at the same time, it may also reduce the accuracy of the query results. On the other hand, a larger value of  $\epsilon$  provides a lower level of privacy protection, but it can also result in more accurate query results. The privacy parameter  $\epsilon$  is used to control the amount of random noise that is added to the query results. The amount of noise added to the query results is directly proportional to the value of  $\epsilon$ . Therefore, a smaller value of  $\epsilon$  results in less noise being added to the query results, which provides a higher level of privacy protection but may also result in less accurate query results. Similarly, a larger value of  $\epsilon$  results in more noise being added to the query results, which provides a lower level of privacy protection but may also result in more accurate query results. It is important to choose an appropriate value of  $\epsilon$  based on the specific privacy and accuracy requirements of the query. In general, a smaller value of  $\epsilon$  is recommended for queries that require a higher level of privacy protection, such as queries that involve sensitive personal information. On the other hand, a larger value of  $\epsilon$  may be acceptable for queries that involve less sensitive information and require more accurate results.



## Exercise B

The protection of individuals' privacy has become a significant concern in data analysis, particularly when the data contains sensitive information. Differential privacy is a widely accepted privacy framework that ensures that the data analysis outputs are independent of the presence or absence of any individual's information in the data set. In this report, we will perform a series of steps to demonstrate the application of differential privacy in protecting individuals' privacy while preserving data utility.

We will use a public dataset and the IBM differential privacy library to execute the following steps:

- Use the Amnesia anonymization tool to apply k-anonymity to the dataset, which will ensure that the individual's information is not identifiable.
- Plot the distribution of numeric features in the dataset using histograms, to understand the data distribution.
- Apply a random noise mechanism to some of the numeric columns using the Gaussian mechanism to protect the individual's privacy while preserving data utility.
- Calculate the differentially private averages for the individuals using the noisy data to obtain statistical insights without violating the individual's privacy.
- Plot the distribution of numeric features after the noise addition and test different values of the  $\epsilon$  parameter to observe the effect of differential privacy on the data utility.

By following these steps, we will illustrate the effectiveness of differential privacy in achieving the balance between data privacy and data utility. Through our analysis, we will gain insights into how the implementation of differential privacy can affect data analysis results and why it is a crucial aspect of data privacy.

### *B1: Use the Amnesia anonymization tool to apply K-Anonymity to the dataset*

In data analysis, anonymization plays a crucial role in protecting individuals' privacy while preserving data utility. The Amnesia anonymization tool is one of the widely used tools for achieving k-anonymity, which ensures that the individuals in a dataset are indistinguishable from each other based on certain attributes.

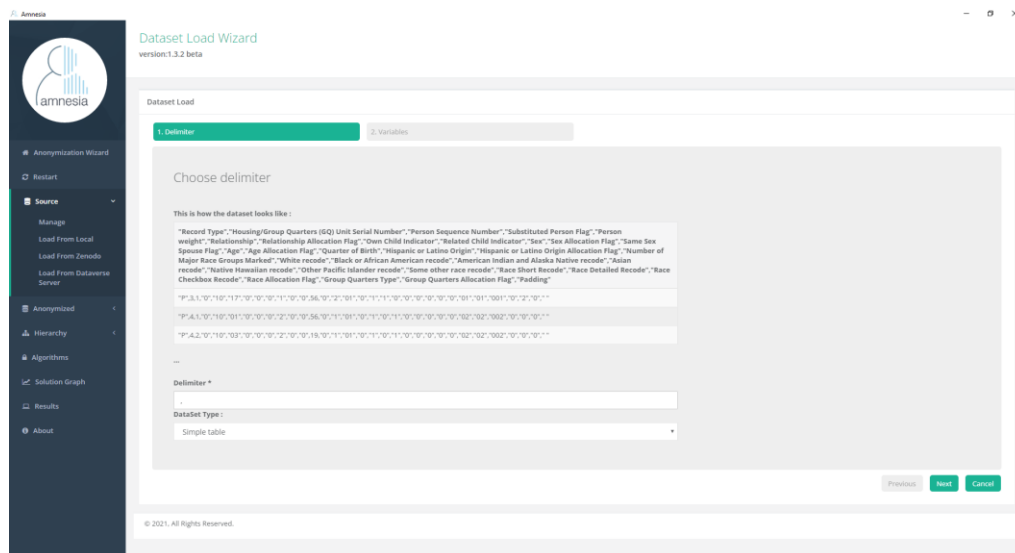
Amnesia is an open-source tool that provides a user-friendly interface for applying different anonymization techniques, including generalization, suppression, and permutation. It can handle various data formats, such as CSV, SQL, and XML, making it versatile for different use cases. Amnesia also provides advanced functionalities, such as privacy models for measuring the privacy risk, quality metrics for evaluating the anonymization results, and validation techniques for verifying the data consistency after the anonymization process.

The main purpose of Amnesia is to provide a comprehensive and customizable solution for data anonymization, which ensures that the anonymized data meets specific privacy and quality requirements. In the following section, we will use Amnesia to apply k-anonymity to our dataset and observe the resulting anonymized data.

### Load the dataset

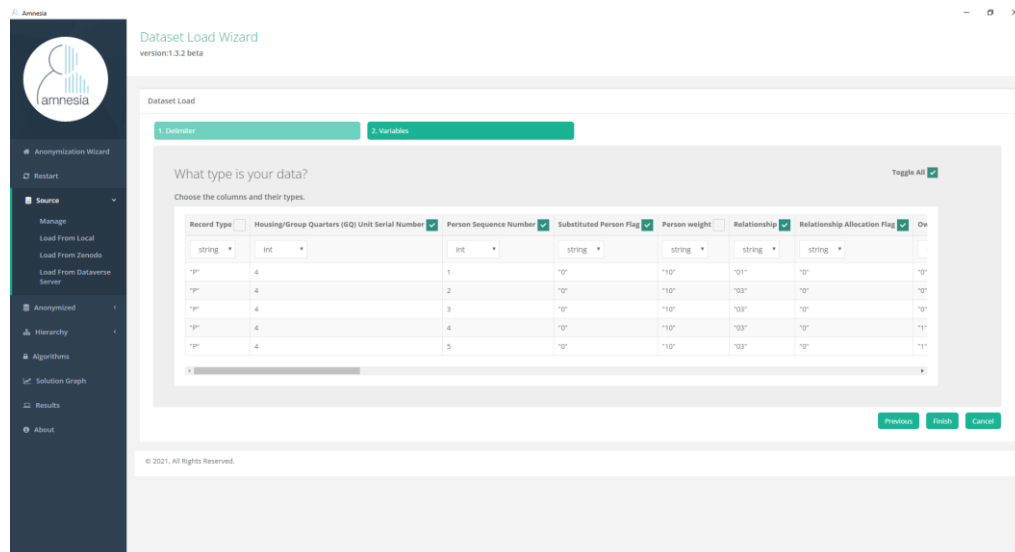
As we delve into the analysis of privacy-preserving techniques, the initial step is to load the relevant dataset into our analysis environment. In this regard, we proceed to load the 'pubs\_df\_titles.txt' file that was previously produced in R from the initial 'de.2010.pums.01.txt' dataset.

To ensure the dataset adheres to the privacy requirements, we utilize the Amnesia anonymization tool, which offers various anonymization techniques such as generalization, suppression, and permutation. To achieve k-anonymity, we apply a **comma delimiter**, and we decide to remove the **'PWEIGHT - Person weight'** and **'RECTYPE - Record Type'** columns, as they contain the same value for all entries.



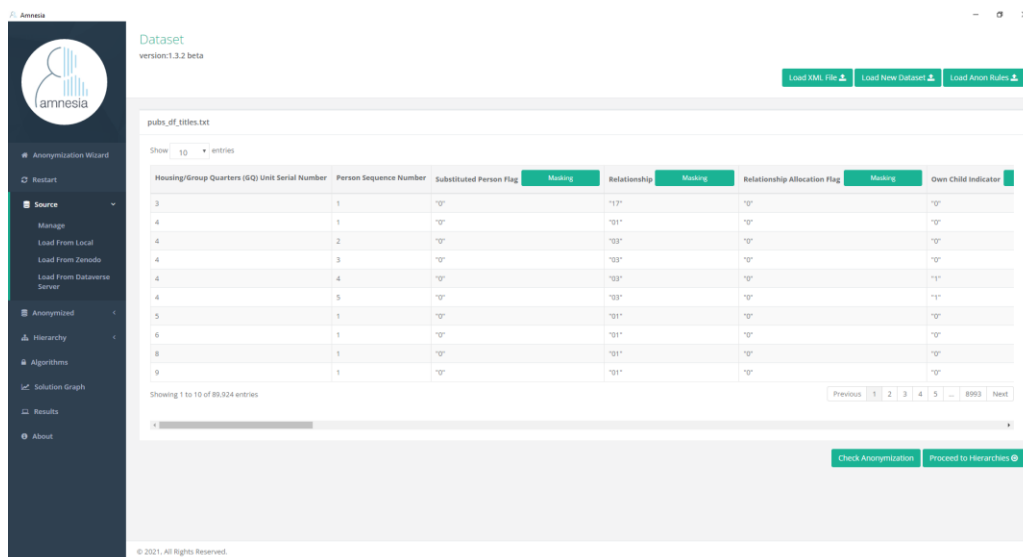
Amnesia Figure 1: Load the data (1/2).

Additionally, we remove 'Padding' from the dataset since it does not add any significant information on the individuals. This process enables us to anonymize the dataset and eliminate any identifiable information, ensuring that individuals' privacy is protected throughout the analysis process.



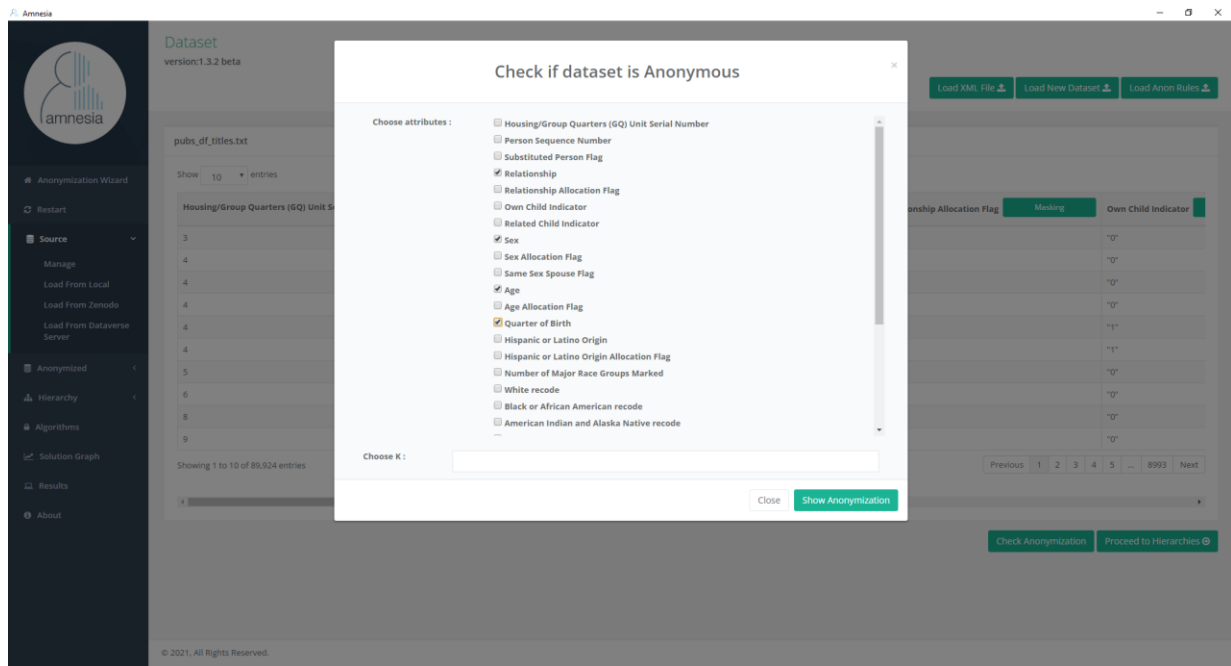
Amnesia Figure 2: Load the data (2/2).

Finally, after we set our options, we load our data successfully and we can preview the loaded dataset.



Amnesia Figure 3: Preview the dataset after the successful load.

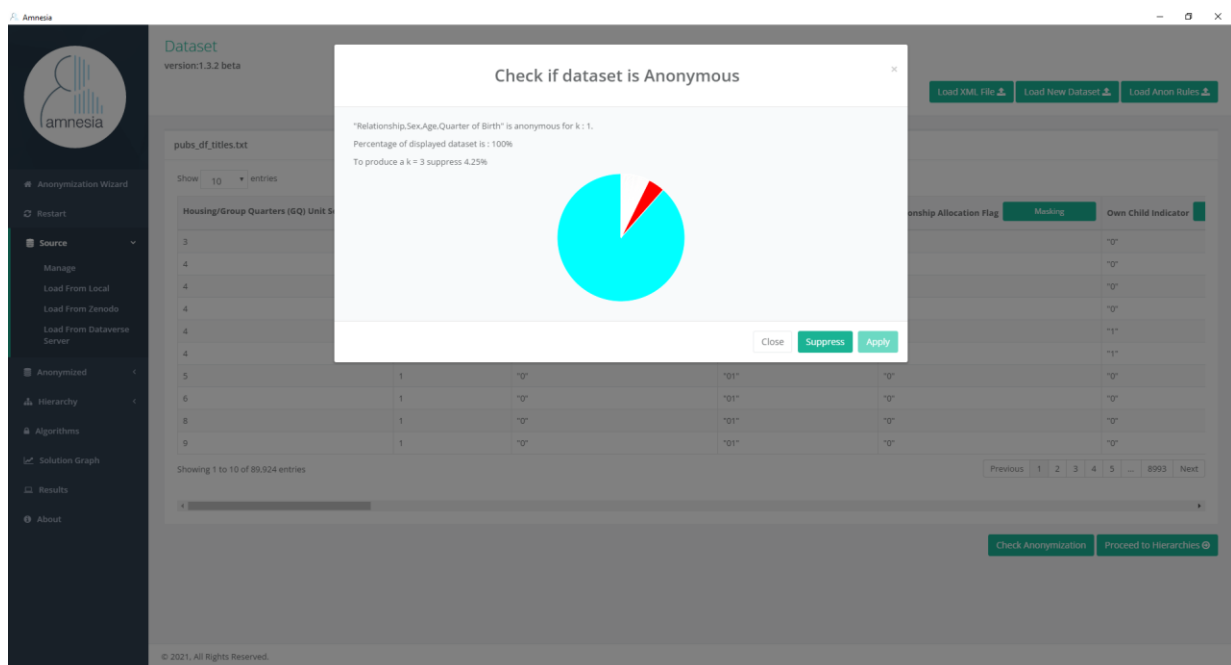
Then, through Amnesia we are able to check if the dataset is Anonymous and their k-anonymity level and the proportion of the dataset that is needed to be suppressed to be able to achieve this level of k-anonymity.



Amnesia Figure 4: Anonymization check.

For our sampling depicted combination, it appears that the dataset has already been anonymized using k-anonymity with a threshold of  $k=1$ . Based on the provided information, the combination of 'Relationship', 'Sex', 'Age', and 'Quarter of Birth' attributes have been anonymized and are anonymous for  $k=1$ , which means that there are at least one other individual in the dataset that share the same values for these attributes.

Furthermore, it is noted that suppressing 4.25% of the dataset would produce a k value of 3, indicating that each combination of 'Relationship', 'Sex', 'Age', and 'Quarter of Birth' attributes would have at least three occurrences in the suppressed dataset. This suggests that the dataset has been carefully anonymized to ensure that individual privacy is protected.



Amnesia Figure 5: 'Relationship', 'Sex', 'Age', and 'Quarter of Birth' combination anonymous check.



## Generalization Hierarchies

Generalization hierarchies refer to the process of categorizing sensitive data into different levels or tiers based on their degree of identifiability. These tiers or levels are designed to ensure that the degree of identifiability of the data is reduced to a level that does not compromise individual privacy. The different levels in an anonymization hierarchy can include:

- **Level 0 (Raw Data):** This level refers to the original data that contains all the raw information without any anonymization or obfuscation. This data is considered to be highly identifiable and sensitive.
- **Level 1 (Generalized Data):** This level involves the application of generalization techniques such as rounding or truncation to reduce the level of detail in the data. This results in data that is less sensitive and less identifiable than the raw data.
- **Level 2 (Aggregated Data):** At this level, data is further anonymized by combining or summarizing it into groups or categories. This reduces the granularity of the data and makes it even less sensitive and less identifiable.
- **Level 3 (Synthetic Data):** This level involves the creation of entirely new data that has similar statistical properties to the original data but does not contain any identifiable information. Synthetic data is generated using machine learning or other statistical methods and can be used for analysis and testing purposes without compromising individual privacy.

By applying anonymization techniques at each level, the degree of identifiability of the data is gradually reduced, ensuring that the data can be used for research and analysis purposes without compromising individual privacy.

For our case, we decide to apply through hierarchies the following categorization in order to combine the anonymization hierarchies algorithms and k-anonymity, two techniques that are commonly used together in data anonymization to ensure that sensitive data is protected from unauthorized access and use, especially for the quasi-identifiers that were mentioned above.

### Housing/Group Quarters (GQ) Unit Serial Number

Grouping the Housing/Group Quarters (GQ) Unit Serial Number to thousands is a method of anonymization that can help to protect the privacy of individuals in a dataset. By reducing the precision of the identifier, it becomes more difficult to identify specific individuals based on their housing or group quarters unit.

In our case, we have grouped the housing/GQ unit serial number into thousands. For example, if the original unit serial number was "25698", we would group it into the "24000 - 25000" category. This reduces the precision of the identifier and makes it more difficult to identify specific individuals based on their housing/GQ unit.

By grouping the housing/GQ unit serial number in this way, we have further simplified the data while still preserving important information about the housing or group quarters units. This can help to protect the privacy of individuals in the dataset while still allowing researchers to perform meaningful analyses on the data. The following picture depicts the above-mentioned hierarchy rules:

Housing/Group Quarters (GQ) Unit Serial Number
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000
41000-42000

Amnesia Figure 6: Housing/Group Quarters (GQ) Unit Serial Number Hierarchy.

To reach this outcome we create the hierarchies file, for example for the housing/GQ unit serial number field we need to group the categories to reduce the precision of the identifier and make it more difficult to identify specific individuals based on this field. In the picture below it is presented this hierarchy text file for this particular field. All hierarchy files for all meaningful 29 variables are part of the deliverable and you can have a look at them.

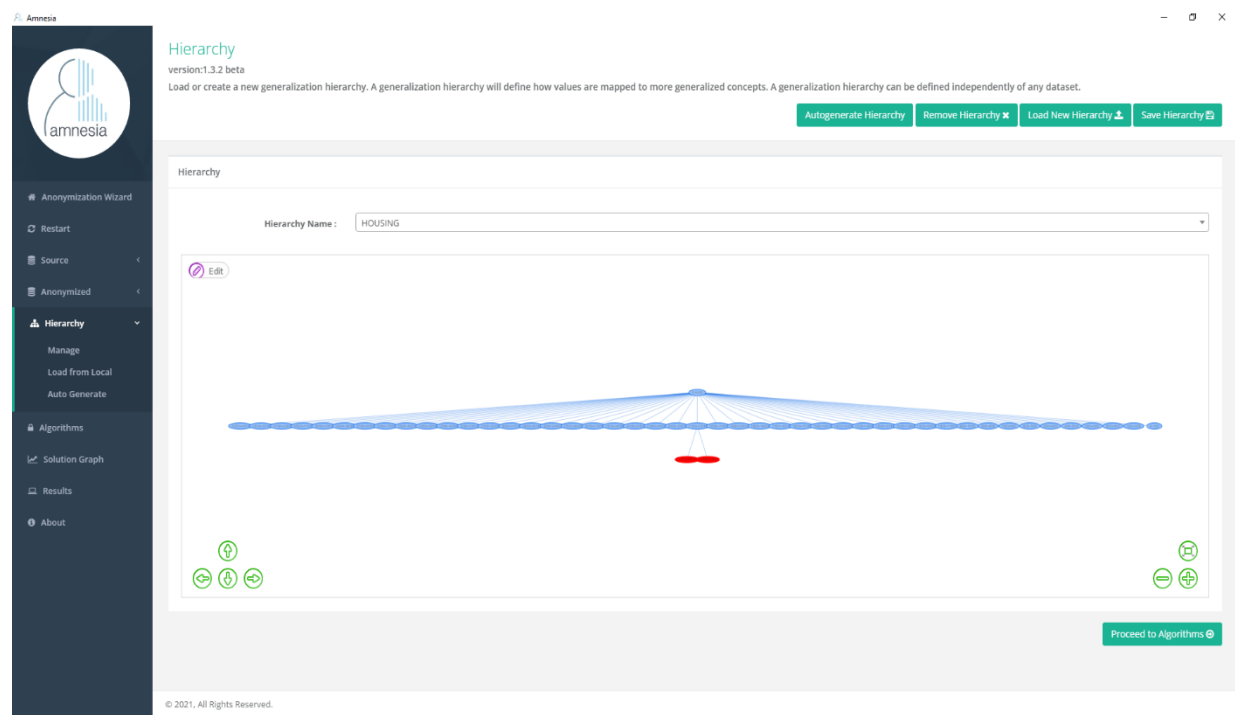
HousingGroup Quarters (GQ) Unit Serial Number.txt - Notepad

```
File Edit Format View Help
distict
name HOUSING
type int
height 3

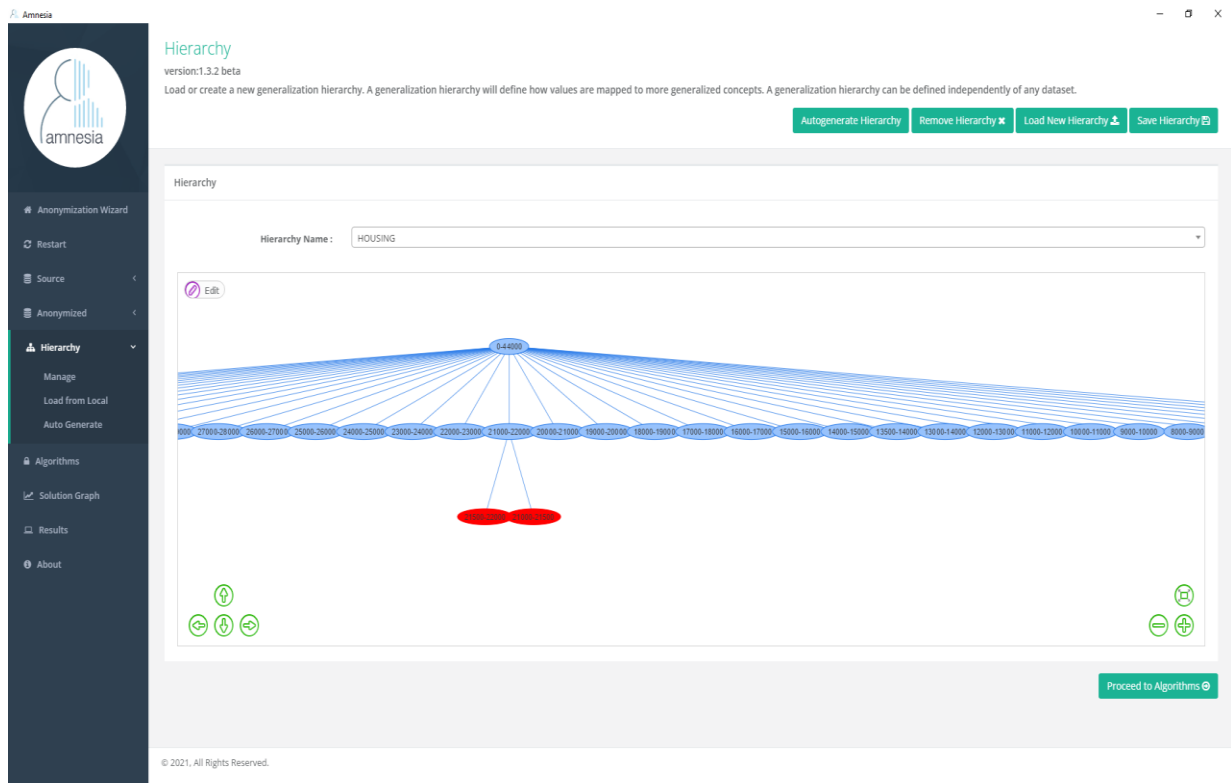
0.0,1000.0 has 0.0,500.0 500.0,1000.0
1000.0,2000.0 has 1000.0,1500.0 1500.0,2000.0
2000.0,3000.0 has 2000.0,2500.0 2500.0,3000.0
3000.0,4000.0 has 3000.0,3500.0 3500.0,4000.0
4000.0,5000.0 has 4000.0,4500.0 4500.0,5000.0
5000.0,6000.0 has 5000.0,5500.0 5500.0,6000.0
6000.0,7000.0 has 6000.0,6500.0 6500.0,7000.0
7000.0,8000.0 has 7000.0,7500.0 7500.0,8000.0
8000.0,9000.0 has 8000.0,8500.0 8500.0,9000.0
9000.0,10000.0 has 9000.0,9500.0 9500.0,10000.0
10000.0,11000.0 has 10000.0,10500.0 10500.0,11000.0
11000.0,12000.0 has 11000.0,11500.0 11500.0,12000.0
12000.0,13000.0 has 12000.0,12500.0 12500.0,13000.0
13000.0,14000.0 has 13000.0,13500.0 13500.0,14000.0
14000.0,15000.0 has 14000.0,14500.0 14500.0,15000.0
15000.0,16000.0 has 15000.0,15500.0 15500.0,16000.0
16000.0,17000.0 has 16000.0,16500.0 16500.0,17000.0
17000.0,18000.0 has 17000.0,17500.0 17500.0,18000.0
18000.0,19000.0 has 18000.0,18500.0 18500.0,19000.0
19000.0,20000.0 has 19000.0,19500.0 19500.0,20000.0
20000.0,21000.0 has 20000.0,20500.0 20500.0,21000.0
21000.0,22000.0 has 21000.0,21500.0 21500.0,22000.0
```

Amnesia Figure 7: Hierarchy Text File for the Housing/Group Quarters (GQ) Unit Serial Number field.

Regarding the Amnesia tool, when we load the hierarchy, text file the graph appears prior to apply it to the algorithms for the fields that we want to apply it. For the Housing/Group Quarters (GQ) Unit Serial Number case, the graph is presented below. We need to point out that this procedure will not be followed for each variable since it is the exact same case, and a single presentation demonstrates the process that it was followed, however you can have a look at the pictures and/or hierarchies folder at the deliverables.

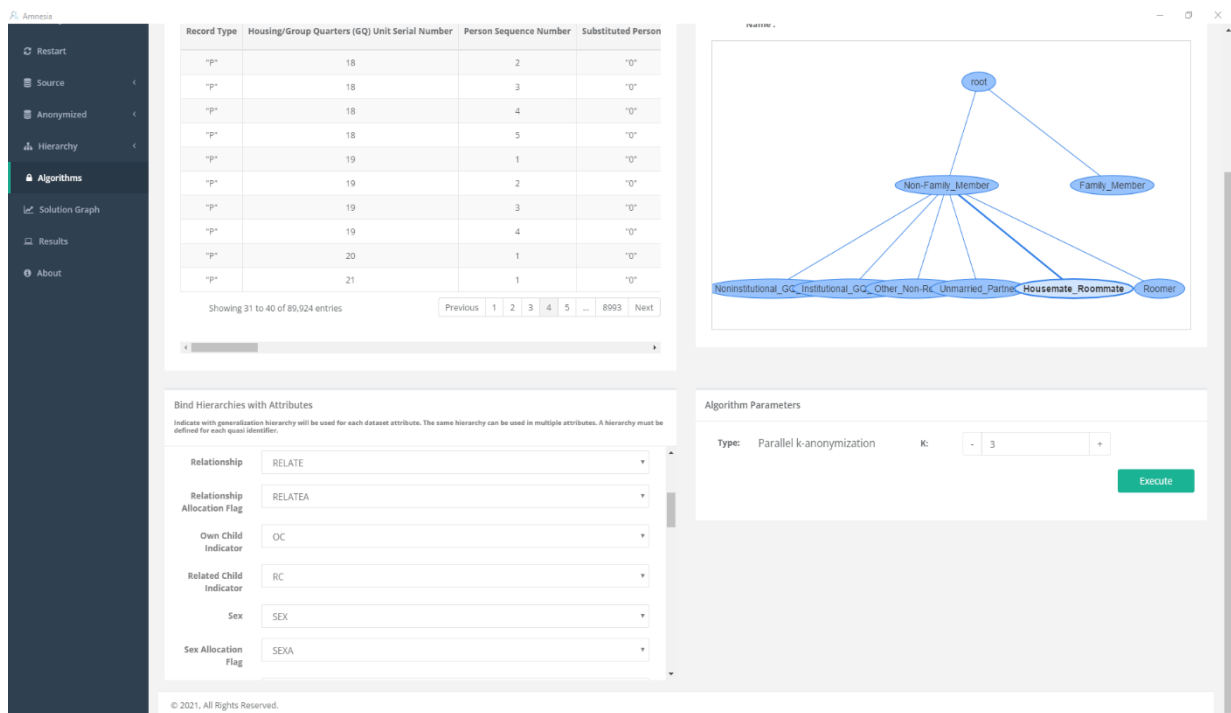


Amnesia Figure 8: Housing/Group Quarters (GQ) Unit Serial Number Amnesia Process (1/2).

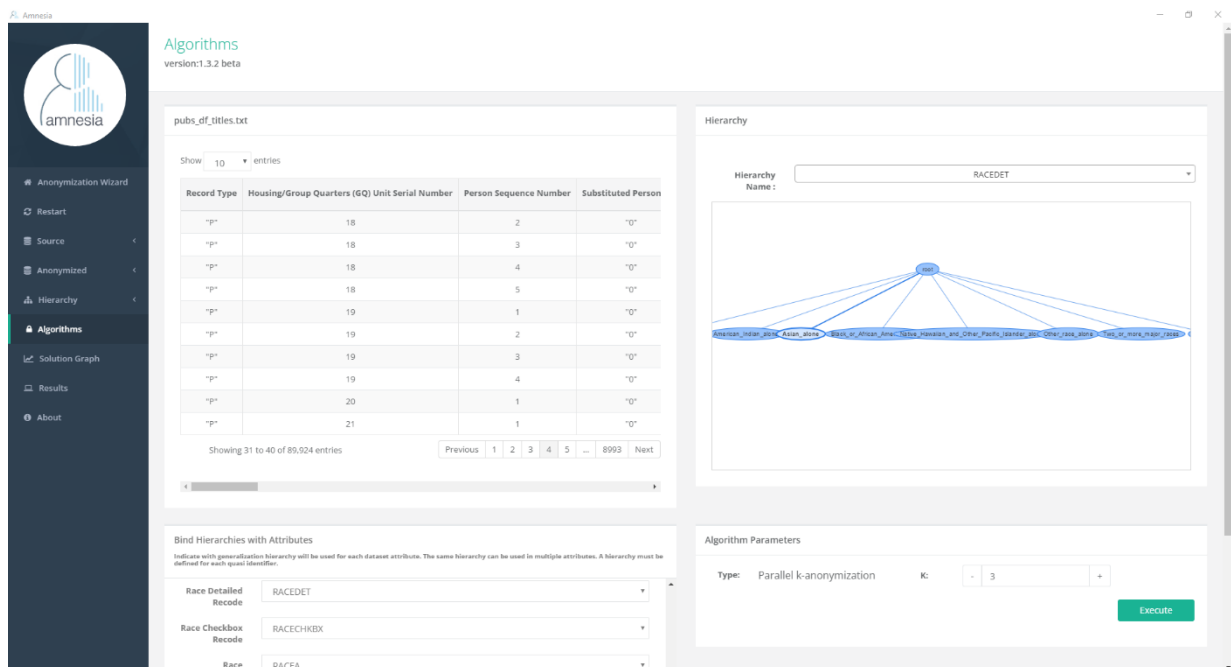


Amnesia Figure 9: Housing/Group Quarters (GQ) Unit Serial Number Amnesia Process (2/2).

The next step is to apply for each particular loaded hierarchy the related algorithm. To be more precise and as you can see below, we set the hierarchies that loaded earlier for each category field in order either to generalized or to improve interpretation for our convenience (e.g. translate 0 to No and 1 to Yes). This methodology was suggested during the lectures as well as on the webinar's videos. On the same note, it was suggested to set the Parallel k-anonymization parameter k equal to 3 - this was also implemented on the demonstration examples, where mentioned that it is a good level of k-anonymity especially with sensitive data.



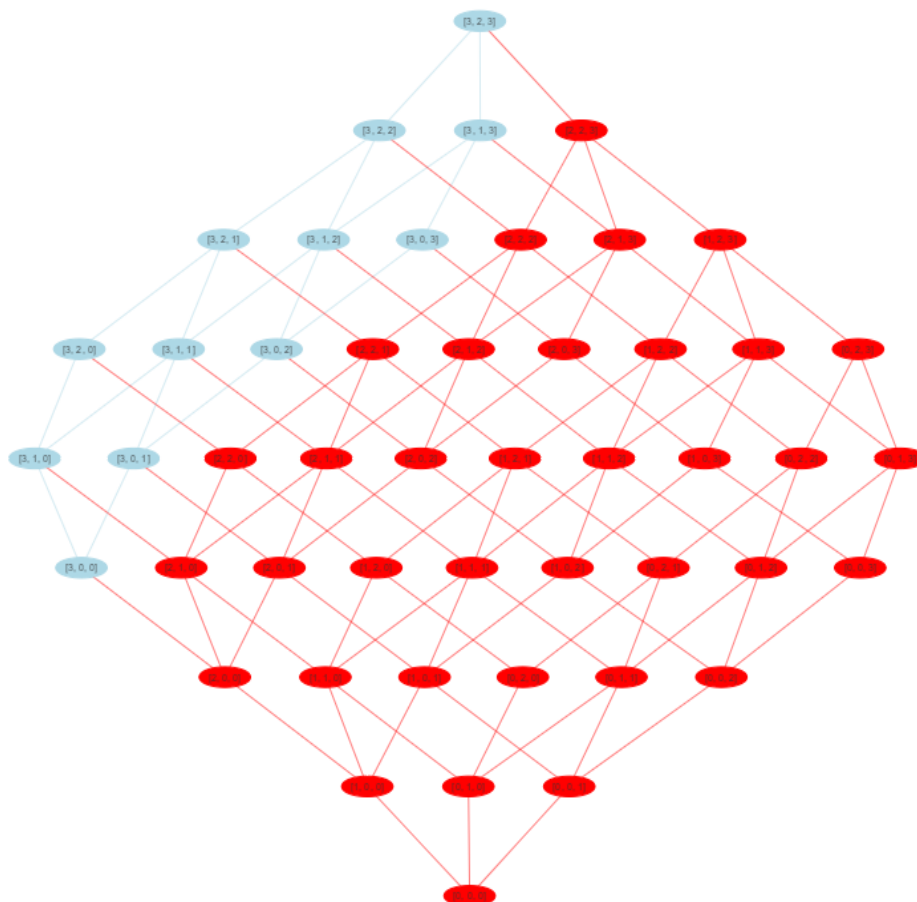
Amnesia Figure 10: Amnesia Algorithms set up based on loaded hierarchies (1/2).



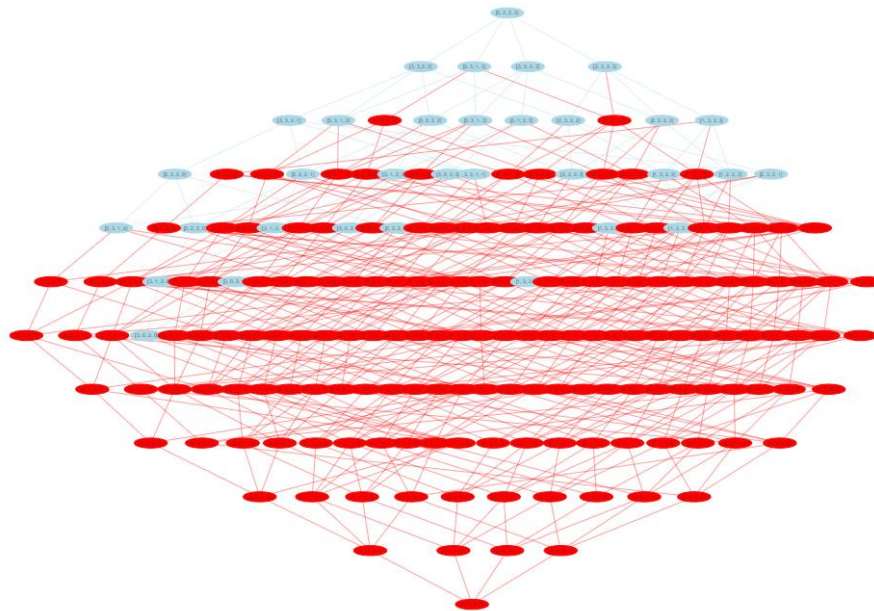
The screenshot shows the Amnesia Algorithms interface. On the left is a sidebar with navigation options: Anonymization Wizard, Restart, Source, Anonymized, Hierarchy, Algorithms (selected), Solution Graph, Results, and About. The main area is titled 'Algorithms version: 1.3.2 beta'. It displays a table of data from 'pubs\_df\_titles.txt' with columns: Record Type, Housing/Group Quarters (GQ), Unit Serial Number, Person Sequence Number, and Substituted Person. The table shows 10 entries, with a total of 89,924 entries. Below the table is a pagination bar. To the right of the table is a 'Hierarchy' section showing a tree diagram for 'RACEDET'. Below the hierarchy is a 'Bind Hierarchies with Attributes' section with dropdowns for 'Race Detailed Recode' (RACEDET), 'Race Checkbox Recode' (RACECHKBX), and 'Race' (RACEA). On the far right is the 'Algorithm Parameters' section, showing 'Type: Parallel k-anonymization', 'K: 3', and an 'Execute' button.

Amnesia Figure 11: Amnesia Algorithms set up based on loaded hierarchies (2/2).

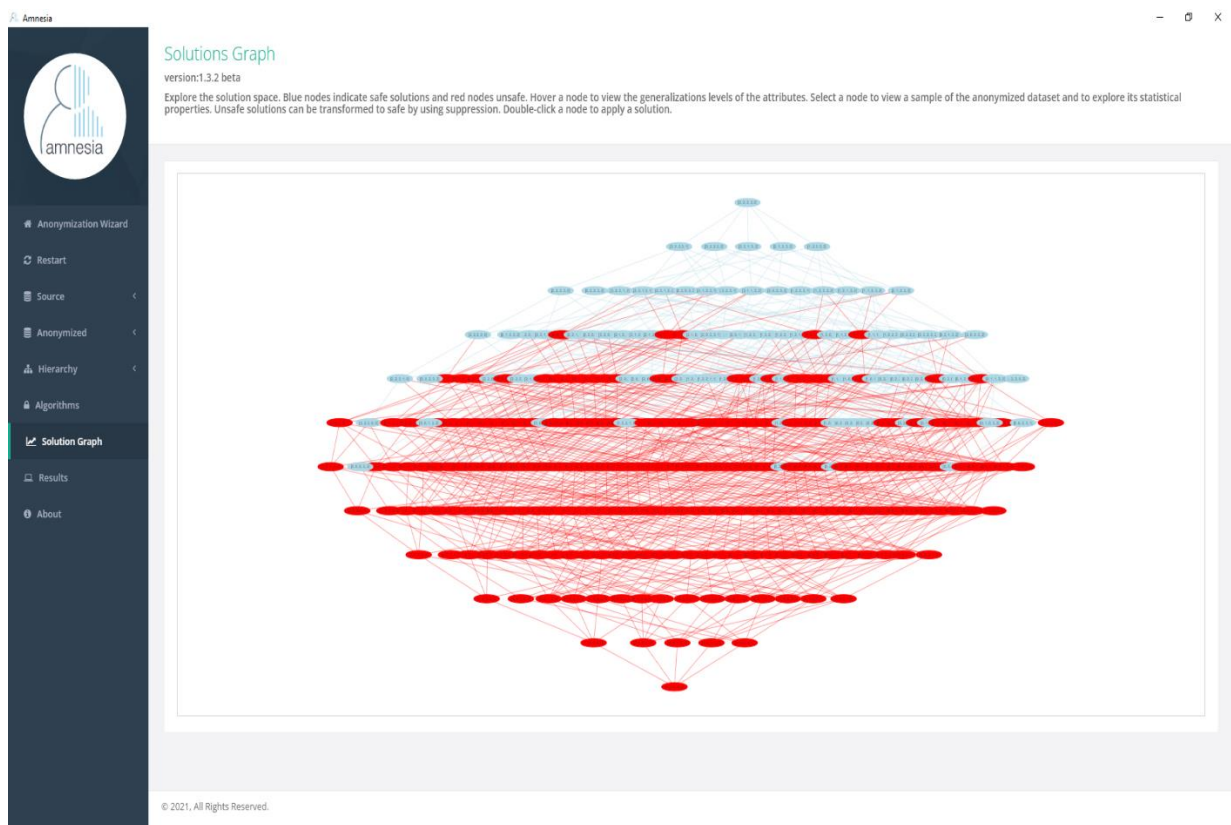
Lastly, we select through the solution graph the level of generalization that we aim for through a vector here a number is related to a particular category field/variable. For better and clarity representation since the solution graph for 29 variables is too complicated there will be presented some simpler solution graphs with three, four, five variables.



Amnesia Figure 12: Solution graph example with 3 variables.



Amnesia Figure 13: Solution graph example with 4 variables.



Amnesia Figure 14: Solution graph example with 5 variables.

After selecting the appropriate vector of generalization and hierarchy level, it is possible to preview the resulting anonymized dataset and obtain various statistics regarding the loss of information, among other factors. Upon switching to the results section, a more detailed comparison can be made between the original dataset and the anonymized version, with the option to export the latter. This process is crucial for maintaining the anonymized dataset and conducting further analyses on various quasi-identifiers and k-anonymity levels.

Subsequently, we will elaborate on the principles guiding the creation of important hierarchies and their formation. This will enable us to compare and contrast different levels of k-anonymity and quasi-identifiers, which will be discussed in a later section.

Finally, a preview of the anonymized dataset results can be observed prior to exporting the final dataset. This allows for a careful examination of the anonymized dataset's characteristics, such as the level of information loss and the effectiveness of the chosen generalization and hierarchy level. Such an examination is necessary for maintaining the anonymized dataset's integrity and accuracy, as well as facilitating further comparisons and analyses on various quasi-identifiers and k-anonymity levels.

Anonymized DataSet

Sex Allocation Flag	Same Sex Spouse Flag	Age	Age Allocation Flag	Quarter of Birth	Hispanic or Latino Origin
NotAllocated	NotChanged	60-80	NotAllocated	First_Semester	NotHispanicorLatino
NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino
NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino
NotAllocated	NotChanged	0-20	NotAllocated	First_Semester	NotHispanicorLatino
NotAllocated	NotChanged	40-60	NotAllocated	First_Semester	NotHispanicorLatino
NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino
NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino
NotAllocated	NotChanged	40-60	Allocated	First_Semester	CentralAmerican
NotAllocated	NotChanged	0-20	Allocated	First_Semester	NotHispanicorLatino
NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino

Statistics

Amnesia Figure 15: Anonymized Dataset preview prior to exportation (1/2).

Anonymized DataSet

Person Sequence Number	Substituted Person Flag	Relationship	Relationship Allocation Flag	Own Child Indicator
3-5	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	Yes
3-5	NotSubstituted	Family_Member	NotAllocated	Yes
3-5	NotSubstituted	Family_Member	NotAllocated	Yes
0-3	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	No
0-3	NotSubstituted	Family_Member	NotAllocated	No

Statistics

Amnesia Figure 16: Anonymized Dataset preview prior to exportation (2/2).

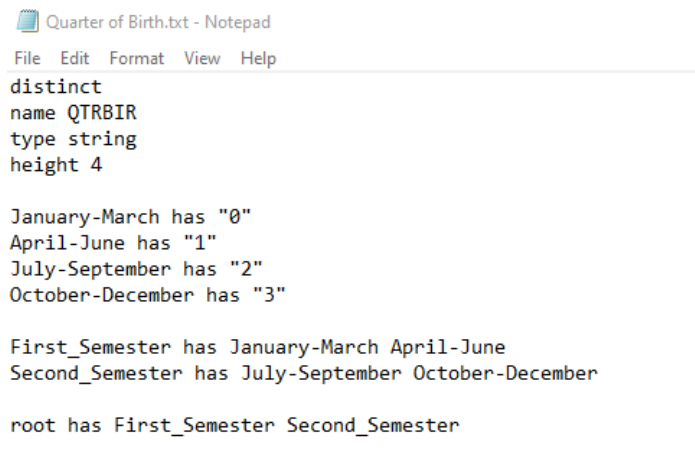


It is imperative to note that the aforementioned results were obtained subsequent to importing all the hierarchies created, coupled with a meticulous setup utilizing the appropriate algorithms and solution graph. Notably, the results were not limited to the Housing/Group Quarters (GQ) Unit Serial Number variable, but encompassed a comprehensive analysis of the anonymized dataset, the appropriate selection of hierarchies and generalization vectors through the solution graph. Overall, such an approach is vital for ensuring the accuracy and reliability of the anonymized dataset, and for conducting thorough comparisons and evaluations of various quasi-identifiers and k-anonymity levels.

### Quarter of Birth

The process of grouping quarter of birth categories from four variables to two through a hierarchical approach is a crucial method of anonymization that effectively protects the privacy of individuals in a dataset. This is accomplished by reducing the number of distinct categories, thereby making it more difficult to identify specific individuals based on their personal information.

One such hierarchy that could be used for this purpose is the QTRBIR hierarchy, which comprises four distinct categories denoted by the quarters in which individuals were born. By grouping these categories into two semesters based on the first and second quarters, individuals' identities can be protected through the use of hierarchies and generalization of grouping. Such an approach enables the anonymized dataset to be used for various research and analytical purposes while safeguarding the privacy and confidentiality of individuals' personal information.



```
distinct
name QTRBIR
type string
height 4

January-March has "0"
April-June has "1"
July-September has "2"
October-December has "3"

First_Semester has January-March April-June
Second_Semester has July-September October-December

root has First_Semester Second_Semester
```

*Amnesia Figure 17: Quarter of Birth Hierarchy Text File.*

### Race (Race Short Recode, Race Detailed Recode, and Race Checkbox Recode)

Anonymizing race categories in a dataset is an important step in ensuring the privacy and confidentiality of individuals. One effective method of anonymization is through hierarchical grouping of race categories.

The RACEDET hierarchy (same procedure followed for all Race Categories - RACESHORT, RACEDET, and RACECHCKBX) can be used for this purpose, comprising of eight distinct categories that include White alone, Black or African American alone, American Indian or Alaska Native alone, Asian alone, Native Hawaiian and Other Pacific Islander alone, Some other race alone, and Two or more major race groups. In this hierarchy, American Indian and Alaska Native alone are grouped together, as are Native Hawaiian and Other Pacific Islander alone, as these groups are often categorized together in demographic data.

By grouping the original race categories into a smaller set of distinct categories through this hierarchy, the privacy and confidentiality of individuals can be protected while still allowing for research and analytical purposes. A sample of the race categories' hierarchies is presented below as well as the loaded to Amnesia section.

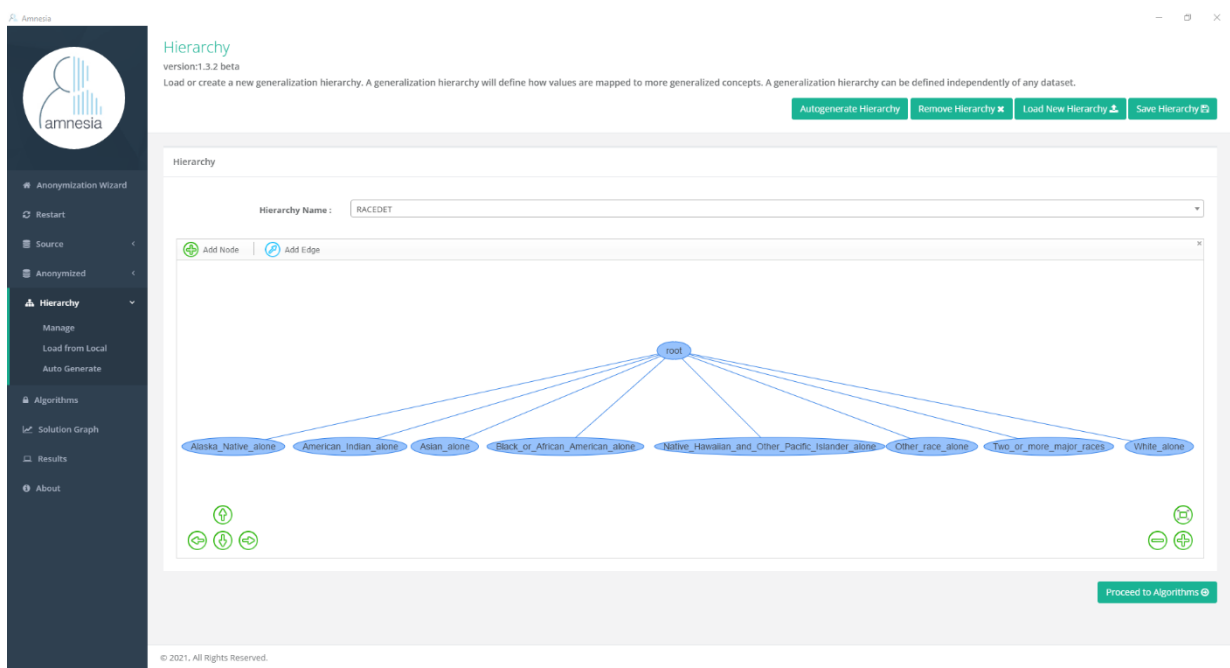
```

Race Detailed Recode.txt - Notepad
File Edit Format View Help
distinct
name RACEDET
type string
height 3

White_alone has "01"
Black_or_African_American_alone has "02"
American_Indian_alone has "03"
Alaska_Native_alone has "32"
Asian_alone has "40"
Native_Hawaiian_and_Other_Pacific_Islander_alone has "63"
Other_race_alone has "73"
Two_or_more_major_races has "74"

root has White_alone Black_or_African_American_alone American_Indian_alone Alaska_Native_alone Asian_alone Native_Hawaiian_and_Other_Pacific_Islander_alone Other_race_alone Two_or_more_major_races
```

Amnesia Figure 18: Race (Race Detailed Recode) Hierarchy Text File.



Amnesia Figure 19: Race (Race Detailed Recode) Hierarchy Loaded to Amnesia Graph.

### Hispanic or Latino Origin

Moreover, anonymizing sensitive data, such as ethnicity, is important to protect the privacy of individuals in a dataset. By grouping similar categories and creating hierarchies, we can reduce the amount of identifying information while still preserving the integrity of the data. This can help to prevent individuals from being targeted or discriminated against based on their personal information. In the case of ethnicity, creating hierarchies can also help to highlight patterns and disparities in health outcomes, education, and other important areas. By anonymizing the data, we can protect privacy and still gain insights into important social issues.

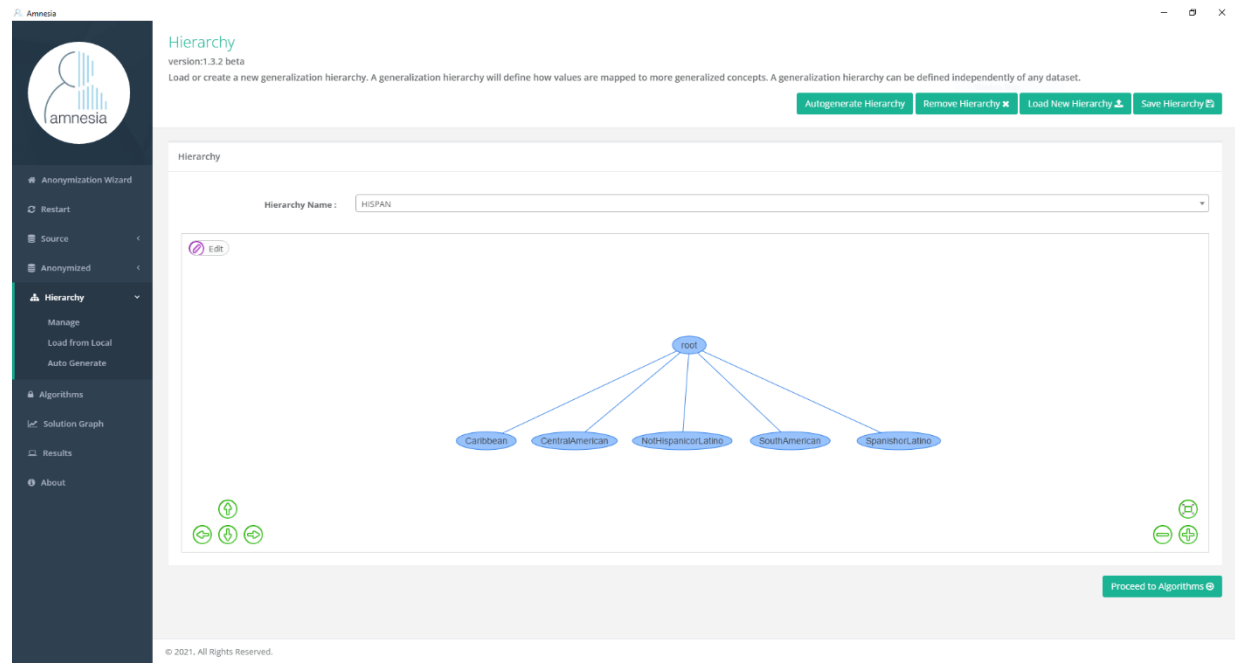
To be more precise, the Hispanic or Latino Origin variable is another example of how anonymization can be achieved through grouping. By grouping the various categories of Hispanic or Latino Origin into broader categories, such as Central American, Caribbean, South American, Spanish or Latino, and Not Hispanic or Latino, it is possible to protect the privacy of individuals in the dataset. By doing so, the specific ethnicities of individuals are not revealed, making it more difficult to identify specific individuals based on their personal information. The hierarchy format used here is helpful in organizing the categories and making it clear how they are related to each other, which can facilitate data analysis while still maintaining privacy. At the same time, with this generalization through hierarchies we group



the 25 initial Hispanic or Latino races to 5 groups based on the topographically origin of each initial category. These 5 categories are:

- Central American,
- Caribbean,
- South American,
- Spanish or Latino,
- Not Hispanic or Latino

Presented below is the hierarchy graph as loaded into Amnesia:



Amnesia Figure 20: Hierarchy graph loaded into Amnesia, representing the Hispanic or Latino Origin variable with the Central American, Caribbean, South American, Spanish or Latino, and Not Hispanic or Latino categories.

### Relationship

The household relationship categories are grouped into two main categories: family members and non-family members. The family members category includes those who have a blood or legal relationship with the householder, such as the householder themselves, their spouse, children, siblings, parents, and other relatives. The non-family members category includes all those who do not have a blood or legal relationship with the householder, such as roomers, boarders, housemates, unmarried partners, and other non-relatives.

It is important to group household relationships in this way because it can help to protect the privacy of individuals. Household relationships can be used as a quasi-identifier, which can be used to link individuals across multiple data sets. For example, if someone knows the relationship between two people in one data set, they may be able to use this information to link these individuals with other data sets, thereby compromising their privacy.

By grouping household relationships into fewer categories, we can reduce the risk of privacy violations. For example, by grouping all family members together, it is more difficult to identify specific individuals based on their relationship to the householder. This can help to protect the privacy of individuals and reduce the likelihood of re-identification attacks.

Therefore, hiding the exact relationship of an individual in a household is important for ensuring that the data is anonymized and protected. The principle of k-anonymity requires that any individual in a data set should be indistinguishable from at least k-1 other individuals, where k is a predetermined

parameter. Grouping household relationships in this way can help to satisfy this principle and reduce the risk of re-identification attacks.

Below it follows the hierarchy text file for household relationship categories, where family members are grouped together into one category and non-family members are grouped into another category and the hierarchy graph when loaded into Amnesia, which shows the distinct values for the "RELATE" field and how they are organized into two main groups: "Family\_Member" and "Non-Family\_Member."

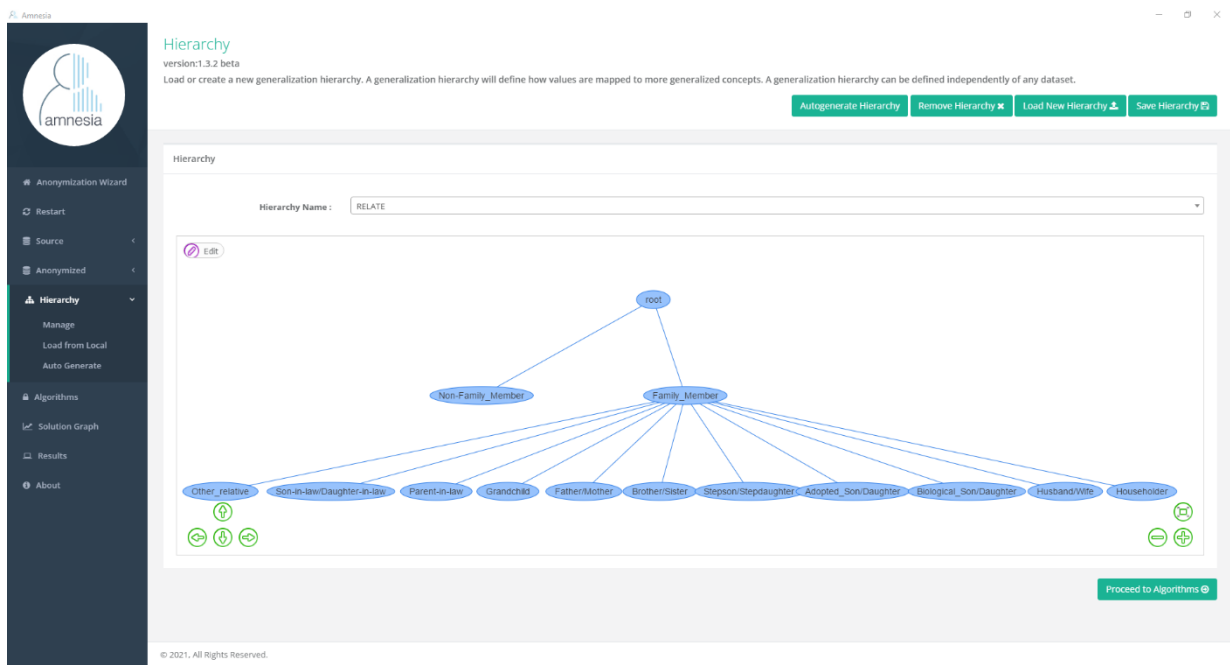
```
Relationship.txt - Notepad
File Edit Format View Help
distinct
name RELATE
type string
height 4

Householder has "01"
Husband/Wife has "02"
Biological_Son/Daughter has "03"
Adopted_Son/Daughter has "04"
Stepson/Stepdaughter has "05"
Brother/Sister has "06"
Father/Mother has "07"
Grandchild has "08"
Parent-in-law has "09"
Son-in-law/Daughter-in-law has "10"
Other_relative has "11"
Roomer has "12"
Housemate_Roommate has "13"
Unmarried_Partner has "14"
Other_Non-Relative has "15"
Institutional_GQ_Person has "16"
Noninstitutional_GQ_Person has "17"

Family_Member has Householder Husband/Wife Biological_Son/Daughter Adopted_Son/Daughter Stepson/Stepdaughter Brother/Sister Father/Mother Grandchild Parent-in-law Son-in-law/Daughter-in-law Other_relative
Non-Family_Member has Roomer Housemate_Roommate Unmarried_Partner Other_Non-Relative Institutional_GQ_Person Noninstitutional_GQ_Person

root has Family_Member Non-Family_Member
```

Amnesia Figure 21: Hierarchy text file to create the hierarchy graph into Amnesia for the Relationship (RELATE) field.



Amnesia Figure 22: Hierarchy graph loaded into Amnesia, representing the Relationship variable with the Family\_Member and Non-Family\_Member categories.

## Age

When it comes to data privacy, age is also considered a quasi-identifier as it can be used in combination with other attributes to identify an individual. Therefore, it is important to separate age into groups to protect individuals' privacy and prevent re-identification attacks.

The hierarchy provided groups age into five categories based on 20-year intervals, ranging from 0 to 100 years old. This grouping allows for a more general representation of age while also maintaining the accuracy of the data. By hiding the exact age of an individual, we are reducing the risk of re-identification attacks that may compromise their privacy.

Additionally, using this hierarchy allows for K-anonymity to be achieved, which is a privacy protection measure that aims to make it difficult for attackers to identify individuals in a dataset. By grouping individuals into larger categories, it ensures that each category has at least K individuals, making it harder for an attacker to pinpoint any one individual within the group. In summary, separating age into groups is important for protecting individuals' privacy and achieving K-anonymity, which can prevent re-identification attacks and ensure data privacy.

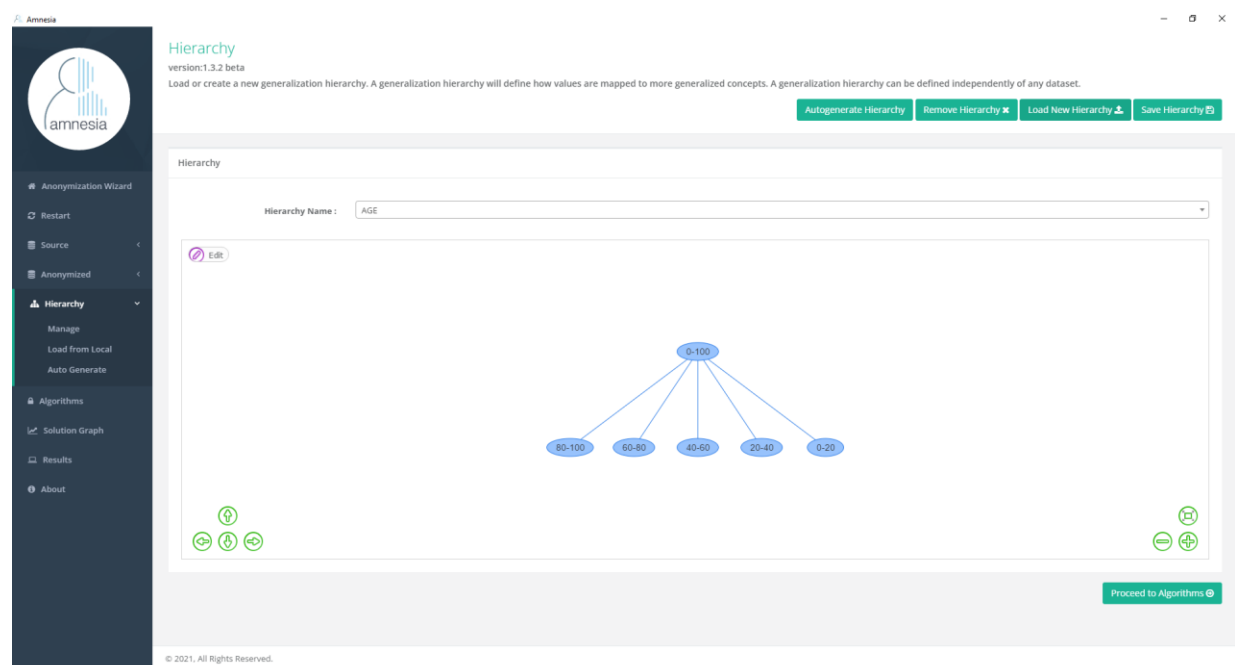
Below are presented the Age hierarchy text and the Amnesia's hierarchy graph for categorizing age ranges into five distinct groups, with each group further divided into subgroups based on intervals of 20 years.

```
Age.txt - Notepad
File Edit Format View Help
distict
name AGE
type int
height 3

0.0,20.0 has 0.0,5.0 5.0,10.0 10.0,15.0 15.0,20.0
20.0,40.0 has 20.0,25.0 25.0,30.0 30.0,35.0 35.0,40.0
40.0,60.0 has 40.0,45.0 45.0,50.0 50.0,55.0 55.0,60.0
60.0,80.0 has 60.0,65.0 65.0,70.0 70.0,75.0 75.0,80.0
80.0,100.0 has 80.0,85.0 85.0,90.0 90.0,95.0 95.0,100.0

0.0,100.0 has 0.0,20.0 20.0,40.0 40.0,60.0 60.0,80.0 80.0,100.0
```

Amnesia Figure 23: The Age hierarchy text with five categories based on 20-year intervals.



Amnesia Figure 24: Hierarchy graph loaded into Amnesia, representing the Age variable with five categories based on 20-year intervals.

## Person Sequence Number

The existence of a hierarchy for a quasi-identifier like Person Sequence Number (PNUM) is important for preserving privacy in data sets. Quasi-identifiers are pieces of information that, when combined, can potentially identify an individual. In this case, PNUM is a quasi-identifier that could potentially be combined with other quasi-identifiers such as age, gender, and occupation to identify an individual.

By having a hierarchy for PNUM, we can group similar values together, which can help protect the privacy of individuals in the data set. For example, if we have a data set that includes information on individuals' ages, occupations, and PNUMs, we can group PNUMs together based on the age ranges they are associated with. This way, if an attacker tries to link PNUMs with other quasi-identifiers to identify individuals, they will have a harder time doing so since multiple PNUMs will be grouped together in the same age range.

In addition, having a hierarchy for PNUM can help protect against the k-anonymity attack. The k-anonymity attack is when an attacker tries to identify an individual by looking for unique combinations of quasi-identifiers. By grouping PNUMs together in a hierarchy, we can ensure that each group contains at least k individuals, where k is the minimum number required for a data set to be considered k-anonymous. This way, even if an attacker manages to identify a group of individuals based on their quasi-identifiers, they will not be able to determine which individual within the group corresponds to a specific PNUM.

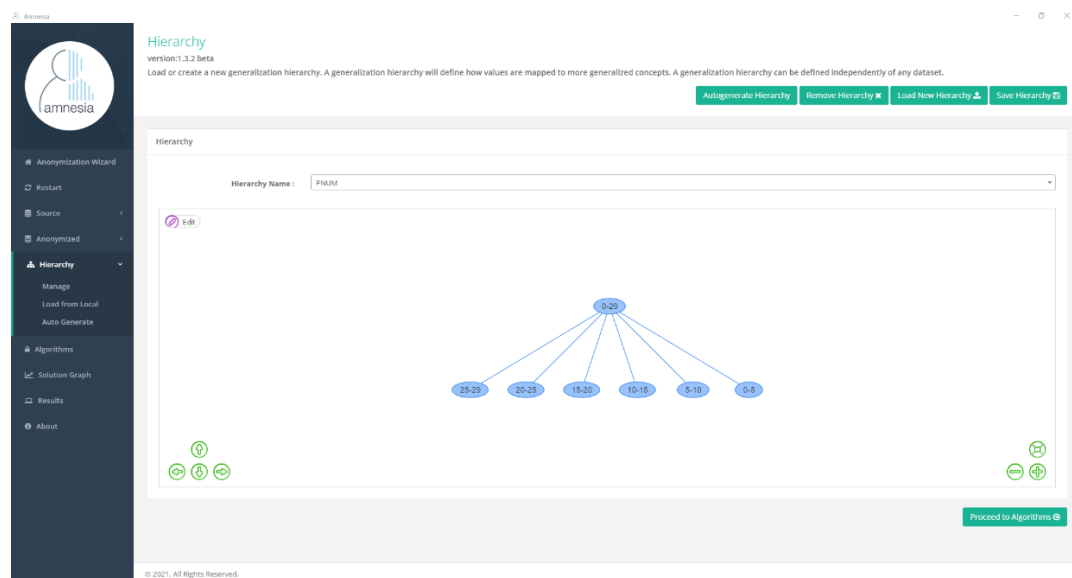
This is a data hierarchy for the Person Sequence Number, with distinct ranges of values divided into sub-ranges. The sub-ranges are defined by the start and end points of each interval, which are shown in the second and third rows of the hierarchy. The ranges go from 0 to 29 and are divided into six sub-ranges. In summary, having a hierarchy for a quasi-identifier like PNUM is important for preserving privacy in data sets. It can help protect against both identity linkage attacks and k-anonymity attacks and can help ensure that individuals' information remains private and secure.

```
Person Sequence Number.txt - Notepad
File Edit Format View Help
distinct
name PNUM
type int
height 3

0.0,5.0 has 0.0,3.0 3.0,5.0
5.0,10.0 has 5.0,8.0 8.0,10.0
10.0,15.0 has 10.0,12.0 12.0,15.0
15.0,20.0 has 15.0,18.0 18.0,20.0
20.0,25.0 has 20.0,23.0 23.0,25.0
25.0,29.0 has 25.0,27.0 27.0,29.0

0.0,29.0 has 0.0,5.0 5.0,10.0 10.0,15.0 15.0,20.0 20.0,25.0 25.0,29.0
```

Amnesia Figure 25: The Person Sequence Number (PNUM) hierarchy text with six subset categories.



Amnesia Figure 26: Hierarchy graph loaded into Amnesia, representing the Person Sequence Number (PNUM) variable with six subset categories.

### Sex and other boolean variables

The variable "Sex" is considered a quasi identifier since it can potentially identify an individual when combined with other information. Therefore, we need to handle it carefully when dealing with privacy-preserving techniques.

One way to protect the privacy of individuals is by creating a hierarchy for the quasi identifier. In this case, we have transformed the "Sex" variable into a numerical value: "1" for male and "2" for female. We also created a hierarchy by defining the categories "Male" and "Female" at the root level.

Having a hierarchy for "Sex" helps to minimize the risk of re-identification by limiting the amount of information that can be disclosed about an individual. For instance, instead of revealing the exact sex of an individual, we can only disclose whether they are male or female.

Furthermore, we have adopted the same approach for all other boolean variables in the dataset. By transforming them into a more representable way and creating hierarchies, we can protect the privacy of individuals while still preserving the utility of the data.

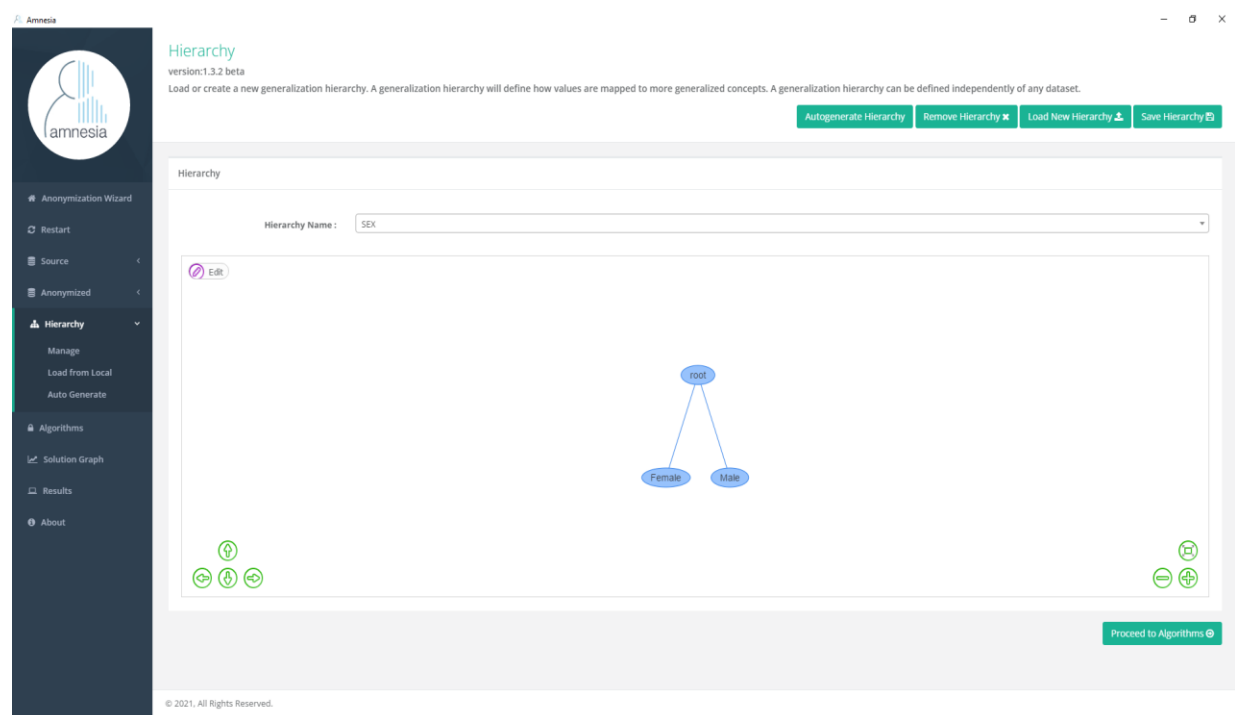
In summary, the existence of a hierarchy for quasi-identifiers such as "Sex" is crucial for privacy-preserving techniques. It helps to limit the amount of information that can be disclosed about an individual and reduce the risk of re-identification.

```
Sex.txt - Notepad
File Edit Format View Help
distinct
name SEX
type string
height 3

Male has "1"
Female has "2"

root has Male Female
```

Amnesia Figure 27: The Sex hierarchy text with two categories.



Amnesia Figure 28: Hierarchy graph loaded into Amnesia, representing the Sex and the other Boolean variables with only two categories.

## Compare the Quasi-Identifiers Original vs Anonymized Dataset

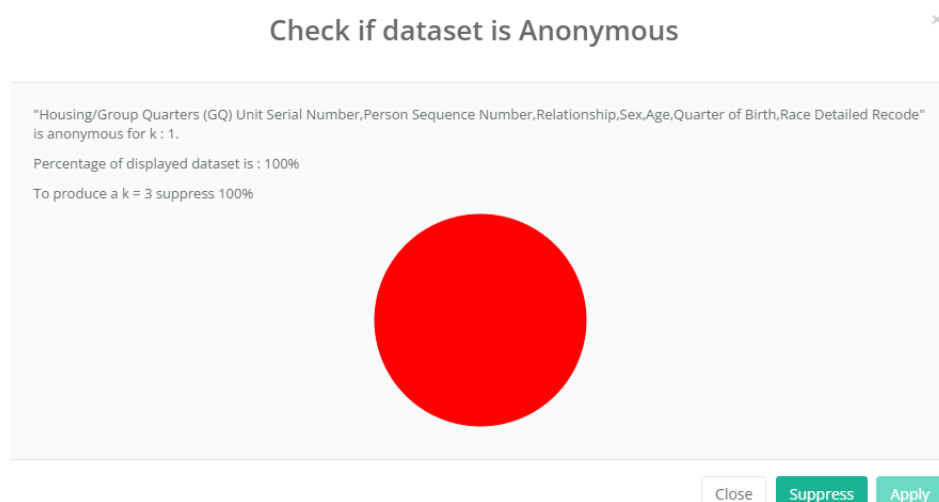
Quasi-identifiers are attributes or a combination of attributes that can potentially identify an individual when combined with external information. Anonymizing data is the process of removing or obscuring information that can be used to identify an individual. Furthermore, anonymizing a dataset to a k-anonymity level of 3 means that each combination of quasi-identifiers occurs at least 3 times in the dataset. This helps to reduce the risk of re-identification by making it harder to single out individuals based on their quasi-identifiers. In addition to the previous point we need to point out that when comparing the quasi-identifiers in an original dataset to an anonymized dataset, there are a few key differences to consider.

In the process of anonymizing a dataset, the presence of quasi-identifiers in the original dataset is a key factor to consider. Anonymization involves either removing or obscuring these quasi-identifiers in order to prevent the identification of individuals. However, this process can have some limitations. The anonymized dataset may have fewer quasi-identifiers than the original dataset, as some may have been removed or combined with other attributes to reduce the risk of identification. Additionally, quasi-identifiers in the anonymized dataset may be less specific or detailed than those in the original dataset, such as age ranges or geographical regions rather than precise ages or addresses. Finally, anonymization techniques such as generalization or perturbation can introduce errors or reduce the precision of the data, potentially leading to less accurate quasi-identifiers in the anonymized dataset. These limitations should be considered when deciding on the appropriate level of anonymization for a dataset, as there is a trade-off between privacy and data accuracy/specificity.

Compared to the original dataset, the anonymized dataset at a **k-anonymity level of 3** is likely to have fewer quasi-identifiers that can be used to identify individuals. This is because the anonymization process typically involves generalizing or grouping quasi-identifiers to ensure that each combination appears at least 3 times in the dataset. This can result in less precise data, with less specific quasi-identifiers. Additionally, the anonymized dataset may still contain some suppressed data, depending on the specific anonymization techniques used. This is because some quasi-identifiers may be too unique or rare to be generalized or grouped effectively, and may need to be suppressed to achieve the desired k-anonymity level.

Overall, while the anonymized dataset at a k-anonymity level of 3 may reduce the risk of re-identification, it may also sacrifice some level of data accuracy and specificity compared to the original dataset. The trade-off between privacy and data accuracy/specificity should be carefully considered when deciding on the appropriate level of k-anonymity for a given dataset. Finally, after careful consideration of the guidance and recommendations provided both in class and during the webinar, the k-anonymity level for the dataset was ultimately selected. This decision was made after taking into account the various instructions and suggestions presented, in order to ensure that the anonymized dataset achieves an appropriate balance between privacy and data accuracy.

The initial dataset achieved the following results.

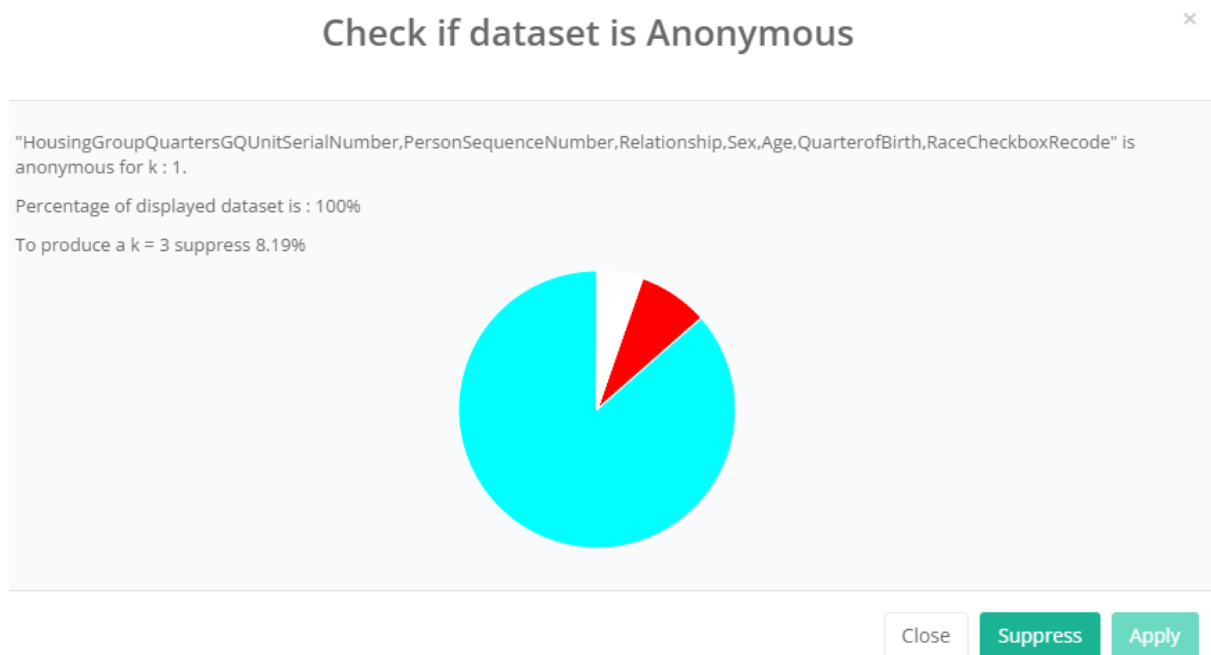


Amnesia Figure 29: Initial Dataset K-Anonymity Validation with 7 Quasi-Identifiers.

Initially, the dataset required a **100% suppression rate** in order to achieve a **k-anonymity level 3**, which would have resulted in significant loss of data. However, through the use of hierarchical generalization techniques on the seven quasi-identifiers, which was highlighted in the webinar that is a big group of quasi-identifiers the dataset was able to achieve the **same k-anonymity level with only an 8.19% suppression rate** (see the picture below). This is a relatively small percentage considering the size of the dataset, which contains approximately 90,000 observations and a group of seven variables. The use of hierarchies allowed for the effective anonymization of the quasi-identifiers while minimizing the amount of suppressed data, thereby preserving the accuracy and usefulness of the dataset.

The significant reduction in suppression rate required to achieve the same k-anonymity level after **implementing hierarchical generalization techniques** that were presented earlier above, is a positive outcome that highlights the importance of these methods in preserving the accuracy and usefulness of datasets while maintaining privacy.

Hierarchical generalization involves grouping similar values of a quasi-identifier into more generalized categories or hierarchies. This approach ensures that the quasi-identifiers are still able to be used for analysis while also minimizing the risk of re-identification. By grouping quasi-identifiers into hierarchies, it becomes more difficult to identify individuals based on their specific attributes, since these attributes have been replaced with more general categories.



*Amnesia Figure 30: : Anonymized Dataset K-Anonymity Validation with 7 Quasi-Identifiers.*

The ability to achieve the same level of k-anonymity with a much lower suppression rate is a clear demonstration of the effectiveness of hierarchical generalization techniques. By minimizing the amount of suppressed data, the usefulness and accuracy of the dataset are preserved.

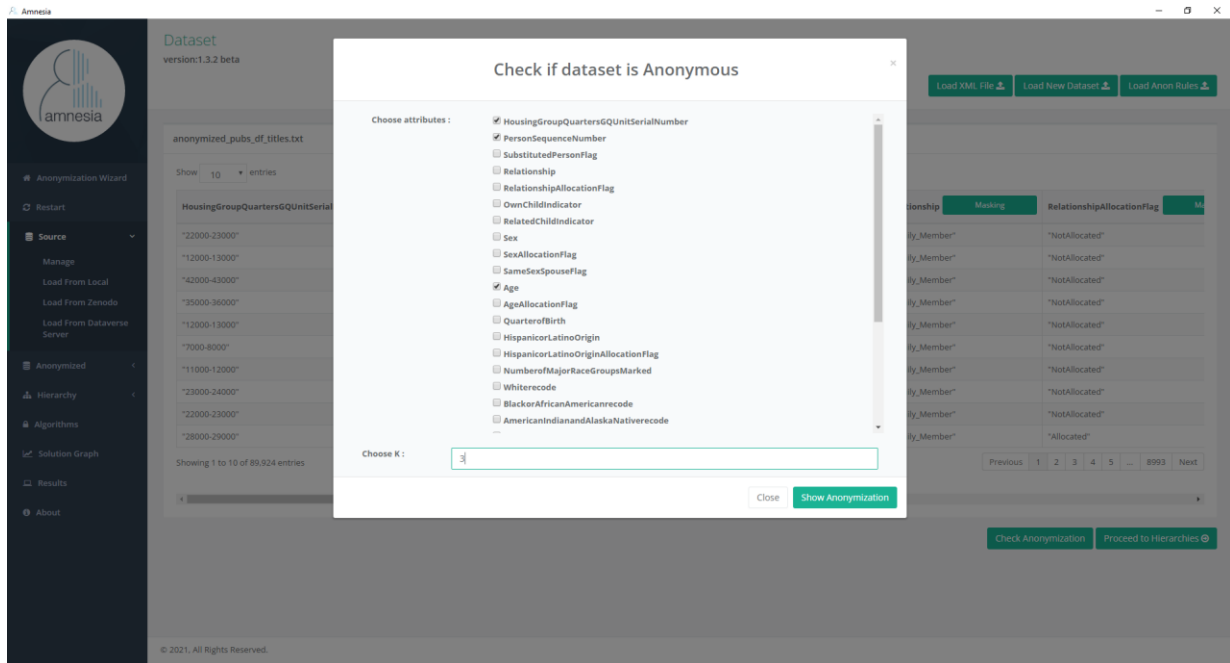
Overall, the results demonstrate the value of implementing hierarchical generalization techniques as an effective and efficient way to maintain privacy in datasets while minimizing the impact on the usefulness and accuracy of the data.

To further illustrate the impact of quasi-identifiers on the anonymization process, we will present two additional examples of quasi-identifier groups and compare the suppression rates required to achieve a k-anonymity level of 3 in both the original and anonymized datasets.



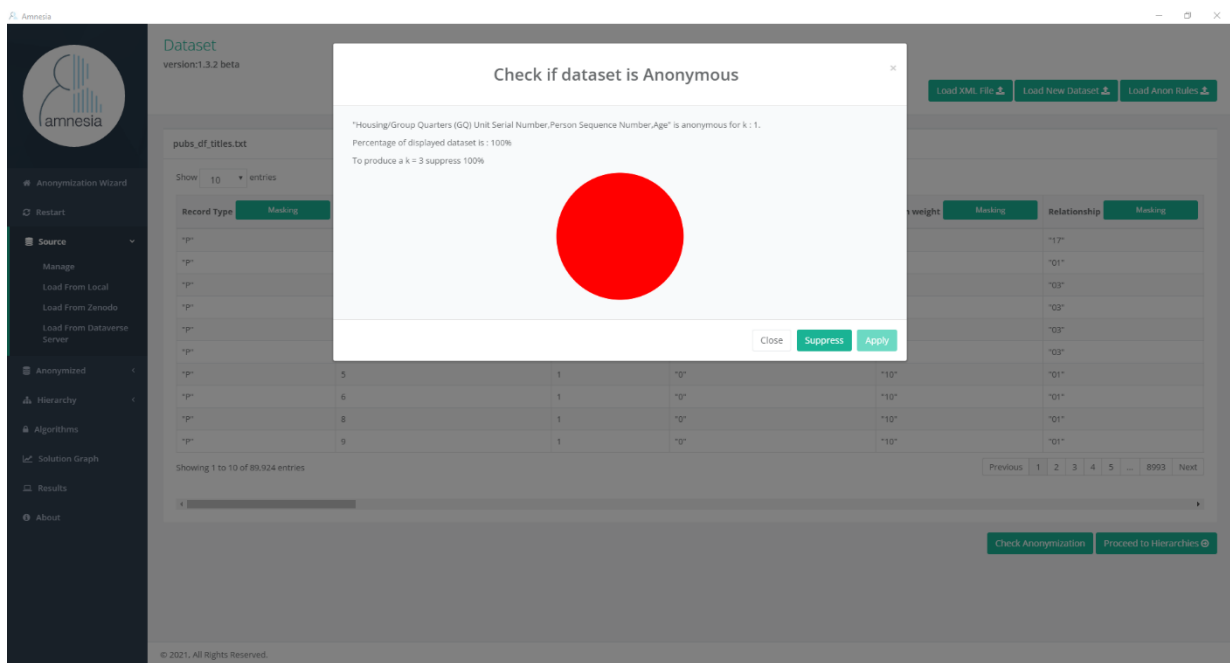
### Example 2: Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number and Age

In this example, we will examine the impact of quasi-identifiers on the anonymization process for a group of variables related to housing and group quarters. Specifically, we will consider the Housing/Group Quarters (GQ) Unit Serial Number, Person Sequence Number, and Age quasi-identifiers. We will compare the suppression rates required to achieve a k-anonymity level of 3 in both the original and anonymized datasets, highlighting the potential trade-offs between privacy and data accuracy/specificity.



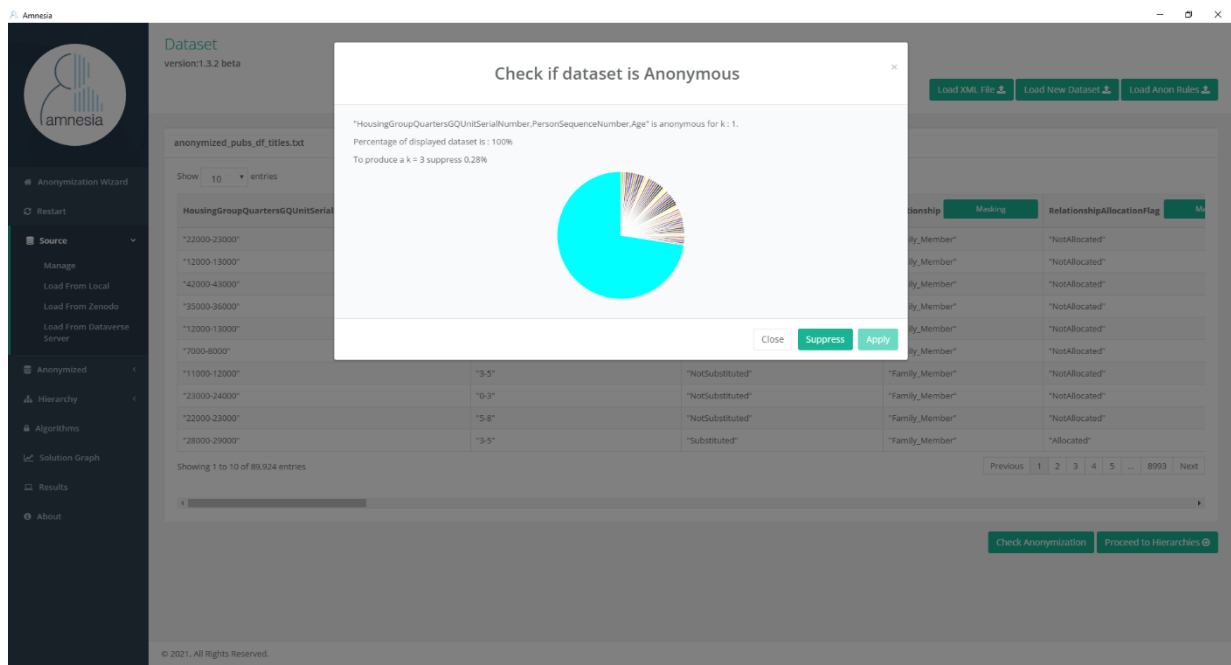
Amnesia Figure 31: Select the quasi-identifiers for the example from the Amnesia's dialog.

After the selection of the 3 quasi-identifiers we can see that the initial's dataset results and compare it with the anonymized dataset results.



Amnesia Figure 32: Initial dataset k-anonymity validation with 3 quasi-identifiers, showing the suppression rate required to achieve a k-anonymity level of 3.



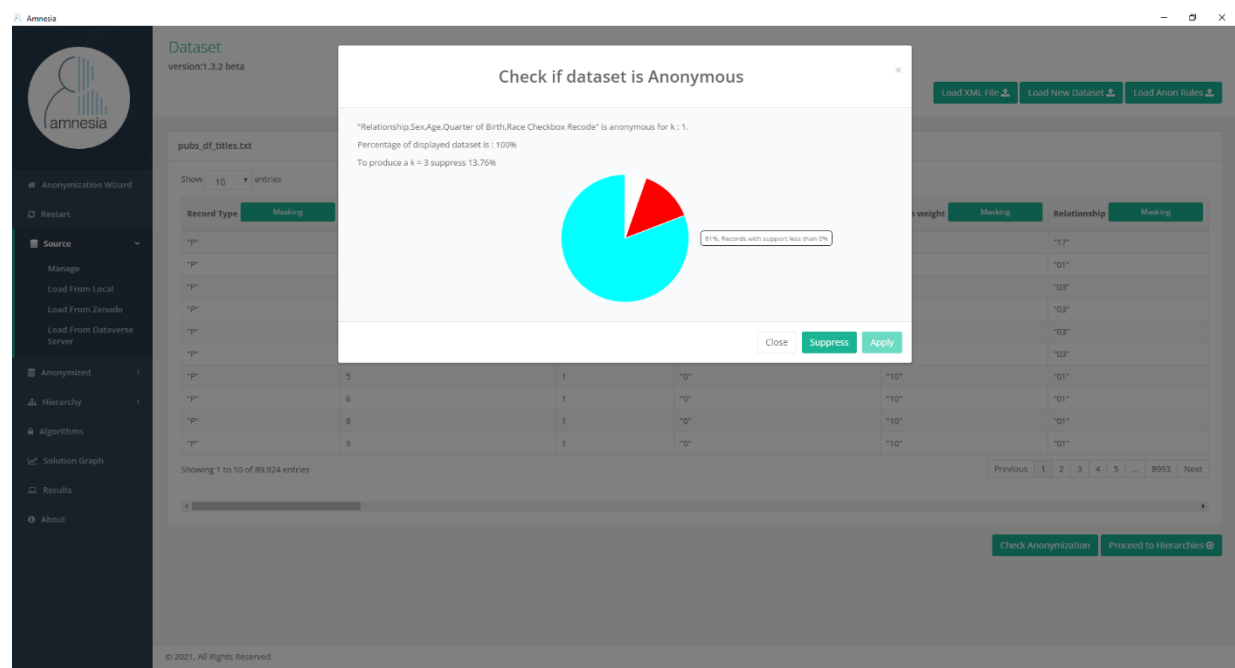


*Amnesia Figure 33: Anonymized dataset  $k$ -anonymity validation with 3 quasi-identifiers, showing the significantly reduced suppression rate required to achieve the same  $k$ -anonymity level of 3 after implementing hierarchical generalization techniques.*

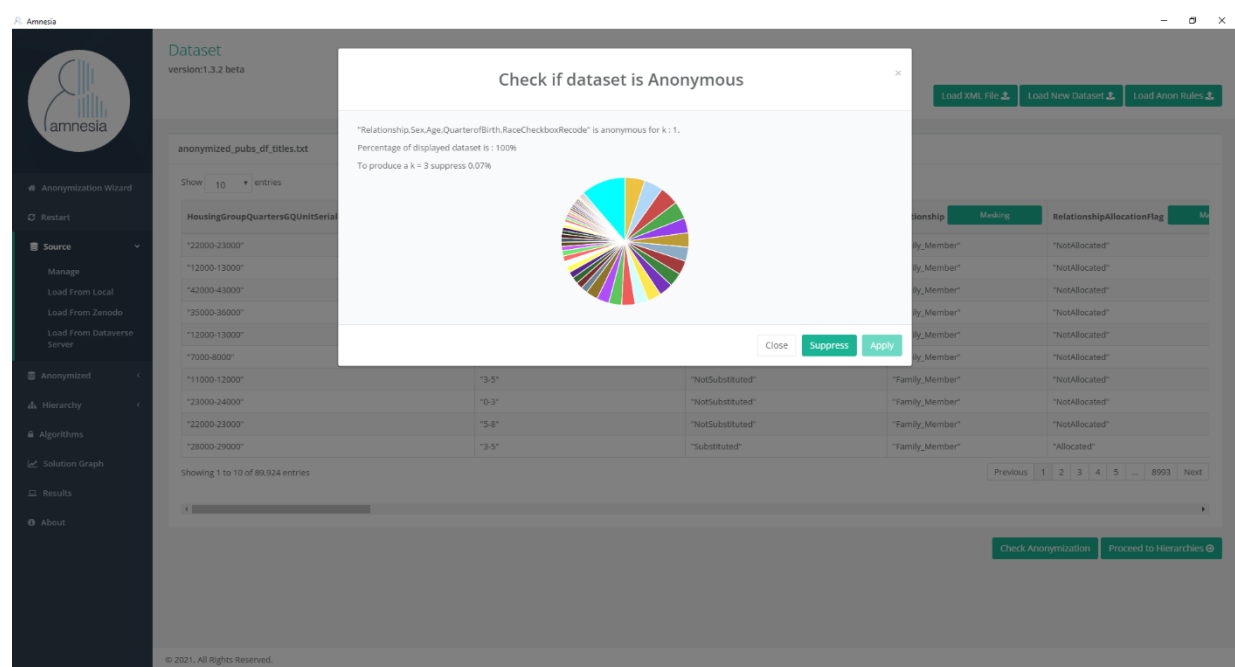
The initial dataset had a **100%** suppress need, which means that every record contained unique combinations of quasi-identifiers, and therefore, all records had to be suppressed to protect the personal identifiable information. However, after anonymization, the suppress need was reduced to **0.28%**, which implies that only a small percentage of the records had unique combinations of quasi-identifiers, and the remaining records were made indistinguishable. Reducing the suppress need in this way is a significant improvement as it allows for the release of a greater portion of the data while still maintaining a high level of privacy. Overall, the results of the anonymized dataset are impressive and an indicator of what we can achieve with the proper use of Generalization Hierarchies and the Amnesia tool.

### Example 3: Relationship, Age, Sex, Quarter of Birth, Race (Race Checkbox Recode)

In this example, we will be discussing a dataset that includes information about individuals' relationships, age, sex, quarter of birth, and race. This type of information can be useful for researchers studying demographics and societal trends. However, as with any dataset that includes personal information, there is a risk of re-identification if the data is not properly anonymized. For instance, imagine you are a 37-year-old private counsellor, and your data is included in this dataset. An attacker could potentially use your relationship status, age, sex, quarter of birth, and race to narrow down the possible identities in the dataset and identify you. To prevent this type of re-identification, data anonymization techniques are needed. By doing so, the dataset can still be used for research purposes while protecting the privacy of individuals whose data is included in the dataset. However, we need to compare the results between the two datasets to prove the abovementioned.



Amnesia Figure 34: : Initial dataset  $k$ -anonymity validation with 5 quasi-identifiers, showing the suppression rate required to achieve a  $k$ -anonymity level of 3.



Amnesia Figure 35: Anonymized dataset  $k$ -anonymity validation with 5 quasi-identifiers, showing the significantly reduced suppression rate required to achieve the same  $k$ -anonymity level of 3 after implementing hierarchical generalization techniques.

The initial dataset required a **13.76%** suppression rate to achieve a k-anonymity level of 3. This means that almost 14% of the original data could not be used in the anonymized dataset because it could lead to individual re-identification. This is a relatively high percentage of suppression and indicates that the initial dataset may have had a significant risk of re-identification.

After anonymization, the suppress rate dropped significantly to **0.07%**. This means that only 0.07% of the data needed to be suppressed to achieve the same k-anonymity level of 3. This is a considerable reduction compared to the initial dataset and suggests that the anonymization process was successful in reducing the risk of re-identification.

Overall, the results suggest that the anonymization process was effective in reducing the amount of data that needed to be suppressed to achieve a desired level of k-anonymity with a very low and very close to 0 percentage of suppression to achieve k-anonymity equal to 3.

In conclusion, it is important to note that there are numerous combinations of quasi-identifiers that can be used in anonymization processes. However, all of these combinations ultimately lead to reduced percentages of suppression needed to achieve a desired level of k-anonymity. This highlights the effectiveness and importance of anonymization techniques in protecting sensitive data, while still allowing for useful insights and analysis to be drawn from the anonymized data

### Final Anonymized Dataset by Amnesia's K-Anonymity Generalization Hierarchies Methods

After applying Amnesia's K-Anonymity Generalization Hierarchies Methods with a k value of 3, we present the final anonymized dataset in csv format below. The anonymization process utilized various techniques such as generalization and grouping to preserve privacy while minimizing data suppression. This anonymized dataset highlights the importance of carefully considering the quasi-identifiers in a dataset and implementing effective anonymization techniques to maintain privacy while preserving data accuracy and usefulness (source file: *final\_anonymized\_dataset.xls*).

AutoSave

FileHomeInsertPage LayoutFormulasDataReviewViewAutomateDeveloperHelpAcrobatTeam

Calibri11A

B I U

Font Color

Clipboard

Wrap Text

General

Conditional Formatting

Format as Table

Normal

Bad

Good

Neutral

Calculation

Check Cell

Insert

Delete

Format

Autosum

Fill

Clear

Sort & Find

Select

Analyze

Sensitivity

Comments

Share

AC21

NotAllocated

	A	B	C	D	E	F	G	H	I	J	K	L	M	N								
	Housing_Group	Quarters_GQ	Unit_Serial	Person_Sequen	Substituted	Person_Flag	Relationship	Relationship_Allocation	Flag	Own_Child_Indic	Related_Child	Indic_Sex	Sex_Allocation	Flag	Same_Sex_Spouse	Flag	Age	Age_Allocation	Flag	Quarter_of_Birth	Hispanic_or_Latino	Original
1	22000-23000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	Second_Semester	NotHispanicorLatino								
2	12000-13000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	First_Semester	CentralAmerican								
3	42000-43000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Female	NotAllocated	NotChanged	0-20	NotAllocated	First_Semester	NotHispanicorLatino								
4	35000-36000	0-3	NotSubstituted	Family_Member	NotAllocated	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	First_Semester	NotHispanicorLatino									
5	13000-13000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Male	NotAllocated	NotChanged	0-20	NotAllocated	First_Semester	NotHispanicorLatino									
6	7000-8000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino								
7	11000-12000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	First_Semester	NotHispanicorLatino								
8	23000-24000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino								
9	12000-23000	5-8	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	20-40	NotAllocated	First_Semester	NotHispanicorLatino								
10	28000-29000	3-5	Substituted	Family_Member	Allocated	No	No	Male	Allocated	NotChanged	20-40	Allocated	First_Semester	NotHispanicorLatino								
11	29000-30000	8-10	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
12	3000-4000	3-5	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Male	NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino								
13	22000-23000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
14	30000-31000	3-5	NotSubstituted	Non-Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	40-60	NotAllocated	First_Semester	NotHispanicorLatino								
15	2000-3000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
16	25000-26000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Male	NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino								
17	21000-22000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino								
18	28000-29000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Female	NotAllocated	NotChanged	0-20	NotAllocated	First_Semester	NotHispanicorLatino								
19	21000-22000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	First_Semester	NotHispanicorLatino								
20	4000-5000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
21	28000-29000	0-3	NotSubstituted	Family_Member	NotAllocated	No	Male	NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino									
22	7000-8000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	First_Semester	NotHispanicorLatino								
23	25000-26000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Female	NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino								
24	0-1000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Male	NotAllocated	NotChanged	0-20	NotAllocated	First_Semester	NotHispanicorLatino								
25	6000-7000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	60-80	NotAllocated	First_Semester	NotHispanicorLatino								
26	22000-23000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	Second_Semester	NotHispanicorLatino								
27	31000-32000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	First_Semester	NotHispanicorLatino								
28	22000-23000	0-3	NotSubstituted	Non-Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	40-60	NotAllocated	First_Semester	NotHispanicorLatino								
29	0-1000	0-3	NotSubstituted	Non-Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	First_Semester	NotHispanicorLatino								
30	37000-38000	5-8	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	60-80	NotAllocated	Second_Semester	NotHispanicorLatino								
31	8000-9000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
32	9000-10000	0-3	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	Second_Semester	NotHispanicorLatino								
33	13000-14000	0-3	NotSubstituted	Non-Family_Member	NotAllocated	No	No	Male	NotAllocated	NotChanged	40-60	NotAllocated	Second_Semester	NotHispanicorLatino								
34	19000-20000	0-3	NotSubstituted	Family_Member	NotAllocated	Yes	Yes	Female	NotAllocated	NotChanged	0-20	NotAllocated	Second_Semester	NotHispanicorLatino								
35	22000-23000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	20-40	NotAllocated	Second_Semester	NotHispanicorLatino								
36	2000-3000	3-5	NotSubstituted	Family_Member	NotAllocated	No	No	Female	NotAllocated	NotChanged	40-60	NotAllocated	Second_Semester	NotHispanicorLatino								

anonymized pubs df titles

### Final Anonymized Dataset 1.

**Microsoft Excel**

File Home Insert Page Layout Formulas Data Review View Automate Developer Help Acrobat Team

Clipboard Font Alignment Number Styles Cells Editing Analysis Sensitivity

Calibri 11 A<sup>x</sup>

B I U [Color Picker] [Background Color]

[Merge & Center]

General Conditional Formatting Format as Table Normal Bad Good Neutral Calculation Check Cell

Insert Delete Format

Σ AutoSum Fill Sort & Filter Find & Select Analyze Data Sensitivity

AC21 NotAllocated

	Q	P	Q	R	S	T	U	V	W	X	Y	Z	AA
	Hispanic_or.Latino_(Number_of.Major.White_race)	Black_or.African.American.In.Asian_race	Native.Hawaiian.Other.Pacific.Some.other.	Race.Short.Records	Race.Detailed.Records	Race.Checkbox.Records	Race.Allocation.Flag						
1	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
2	NotAllocated	One_race	Yes	No	No	No	No	No	Yes	Two_or_more_major_race_groups	Two_or_more_major_races	Two_or_more_major_races	NotAllocated
3	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
4	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
5	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
6	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
7	NotAllocated	One_race	No	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
8	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
9	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
10	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
11	Allocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	Allocated
12	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
13	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
14	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
15	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
16	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
17	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
18	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
19	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
20	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
21	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
22	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
23	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
24	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
25	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
26	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
27	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
28	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
29	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
30	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	Allocated
31	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated
32	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
33	NotAllocated	One_race	No	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
34	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
35	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
36	NotAllocated	One_race	Yes	No	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated
37	NotAllocated	One_race	No	Yes	No	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated

anonymized pubs df titles

### Final Anonymized Dataset 2.

Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	hite.recode	Black.or.African.American.In.Asian.recode	Native.Hawaiian.Other.Pacific.Some.other				Race.Short.Recode	Race.Detailed.Recode	Race.Checkbox.Recode	Race.Allocation.Flag	Group.Quarters.Type	Group.Quarters.Al
2	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
3	Yes	No	No	No	No	No	Two_or_more_major_race_groups	Two_or_more_major_races	Two_or_more_major_races	NotAllocated	Not_in_a_GQ	NotAllocated
4	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
5	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
6	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
7	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
8	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
9	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
10	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
11	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	Allocated	Not_in_a_GQ	NotAllocated
12	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
13	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
14	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
15	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Institutional_GQ	NotAllocated
16	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
17	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
18	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
19	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
20	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
21	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
22	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
23	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
24	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
25	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
26	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
27	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
28	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
29	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
30	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	Allocated	Not_in_a_GQ	NotAllocated
31	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated
32	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
33	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
34	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
35	Yes	No	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
36	No	Yes	No	No	No	No	White_alone	White_alone	White_alone	NotAllocated	Not_in_a_GQ	NotAllocated
37	No	Yes	No	No	No	No	Black_or_African_American_alone	Black_or_African_American_alone	Black_or_African_American_alone	NotAllocated	Not_in_a_GQ	NotAllocated

Final Anonymized Dataset 3.

## Differential Privacy

One of the crucial approaches adopted by major technology companies is to effectively leverage the abundant personal data available online while ensuring the protection of data subjects from various potential risks and harms. Differential privacy (DP) is a robust and rigorous definition of privacy that applies to statistical and machine learning analyses. It serves as a criterion for privacy protection and has spawned numerous tools and techniques that satisfy this standard. The aforementioned diagram illustrates the representation of data information within the framework of DP. In this context, general information refers to data that is not specific to any individual data subject but instead pertains to the population as a whole. Specifically, it encompasses information that pertains to the entire population included in the dataset, rather than any individual or subgroup of data subjects. In contrast, private information denotes data that is specific to individual data subjects.

Differential privacy (DP) offers a robust and mathematically provable guarantee of privacy protection against various types of privacy attacks, such as differencing attacks, linkage attacks, and reconstruction attacks. By utilizing DP, any party observing the results of a differentially private analysis would essentially draw the same conclusions about the private information of any individual, regardless of whether or not that individual's private information was included in the analysis. It is worth noting that Differential Privacy (DP) does not provide an absolute assurance that an individual's perceived secrets or confidential information will remain secure. It is crucial to distinguish between general and private information to effectively leverage the benefits of DP and minimize potential harm. DP guarantees protection only for private information, as stated earlier. Therefore, if the so-called "secret" is classified as general information, it will not be subject to protection under the DP framework.

To be more precise, Differential Privacy (DP) is a powerful framework for the analysis of sensitive personal information and privacy protection, which boasts several valuable properties. Firstly, DP enables the quantification of privacy loss, which serves as a crucial measure in any DP mechanism or algorithm. This allows for comparisons among different techniques, and the controllability of privacy loss ensures a trade-off between it and the accuracy of general information.

Secondly, DP allows for the analysis and control of cumulative privacy loss over multiple computations through composition. Understanding the behavior of differentially private mechanisms under composition enables the design and analysis of complex differentially private algorithms from simpler



building blocks. Additionally, DP permits the analysis and control of privacy loss incurred by groups, such as families, through the concept of Group Privacy.

Lastly, DP is immune to post-processing, which is known as the Closure Under Post-Processing property. This means that a data analyst, without additional knowledge about the private database, cannot compute a function of the output of a differentially private algorithm and make it less differentially private. The combination of these valuable properties has solidified DP's position as a rich and comprehensive framework for privacy-preserving data analysis.

In the context of Differential Privacy (DP) algorithms, two crucial quantities that must be considered are Epsilon ( $\epsilon$ ) and Accuracy. Epsilon is a metric that quantifies the degree of privacy loss that occurs due to a differential change in the data, such as adding or removing a single entry. A smaller value of epsilon indicates better privacy protection. Accuracy, on the other hand, refers to the degree of closeness between the output of DP algorithms and the pure output. In the case of Private Machine Learning with PATE, the classification accuracy on the test set serves as a statistic for evaluating accuracy. Hence, careful consideration of both epsilon and accuracy is critical for designing and evaluating effective DP algorithms.

Regarding the Differential Privacy (DP) algorithms, it is important to note that decreasing the value of epsilon ( $\epsilon$ ) leads to a decrease in accuracy. This trade-off between privacy and accuracy must be carefully considered when designing DP algorithms. It is worth noting that a 0-differential privacy algorithm, while providing strong privacy protection, may have very low accuracy, rendering it practically useless. In such cases, the algorithm may generate output that is dominated by noise, providing little to no meaningful information. Furthermore, it is important to note that when  $\epsilon=0$  (and  $\delta=0$  in the general case), it is equivalent to absolute privacy. This can be derived directly from the definition of Differential Privacy. Specifically,  $\epsilon=0$  implies that  $\Pr[K(D)] = \Pr[K(D')]$ , which leads to the algorithm  $K$  being independent of the data and, thus, providing perfect privacy protection.

### ***Differential Privacy - The Python library Diffprivlib***

Diffprivlib is a comprehensive library designed to facilitate experimentation, investigation, and development of applications in differential privacy. It offers a range of functionalities suitable for various purposes, such as exploring the implications of differential privacy on machine learning accuracy through classification and clustering models, building customized differential privacy applications, and conducting general experimentation with differential privacy techniques. It provides an extensive collection of mechanisms that support the development of differential privacy applications. These mechanisms include algorithms for data anonymization, sanitization, and perturbation, among others. Therefore, diffprivlib is an ideal tool for researchers and practitioners who wish to engage in a broad range of activities relating to differential privacy, from experimentation to application development.

The first component of the library is the Mechanisms module, which serves as the foundation of differential privacy by providing the necessary building blocks used in all models implementing differential privacy. Although mechanisms do not have default settings, they are ideal for use by experts looking to implement their models or for individual investigations.

The second component is the Models module, which contains various machine learning models that incorporate differential privacy. These models are available for clustering, classification, regression, dimensionality reduction, and pre-processing. They enable users to explore the impact of differential privacy on machine learning accuracy using various models.

The third component of Diffprivlib is the Tools module. It includes a range of generic tools for differentially private data analysis. This includes differentially private histograms, which follow the same format as NumPy's histogram function. These tools are useful for performing differentially private data analysis tasks.

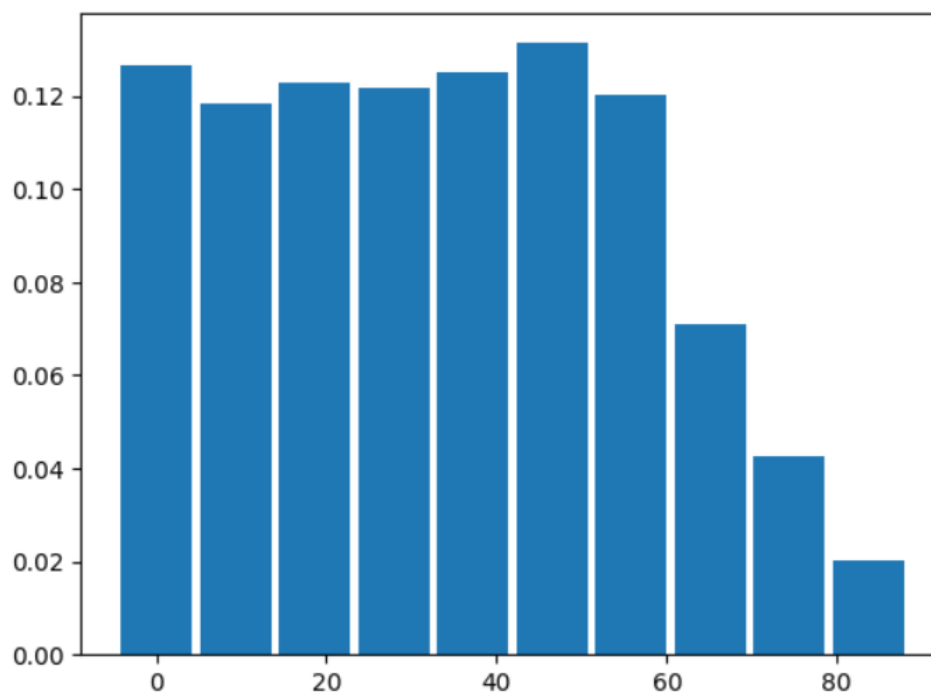
The final component is the Accountant module, which includes the BudgetAccountant class. This class is used to track privacy budget and calculate total privacy loss using advanced composition techniques. This enables users to maintain a comprehensive understanding of the privacy budget utilized in their differentially private analysis.

## B2: Distribution of Numeric Features in the Dataset Using Histograms

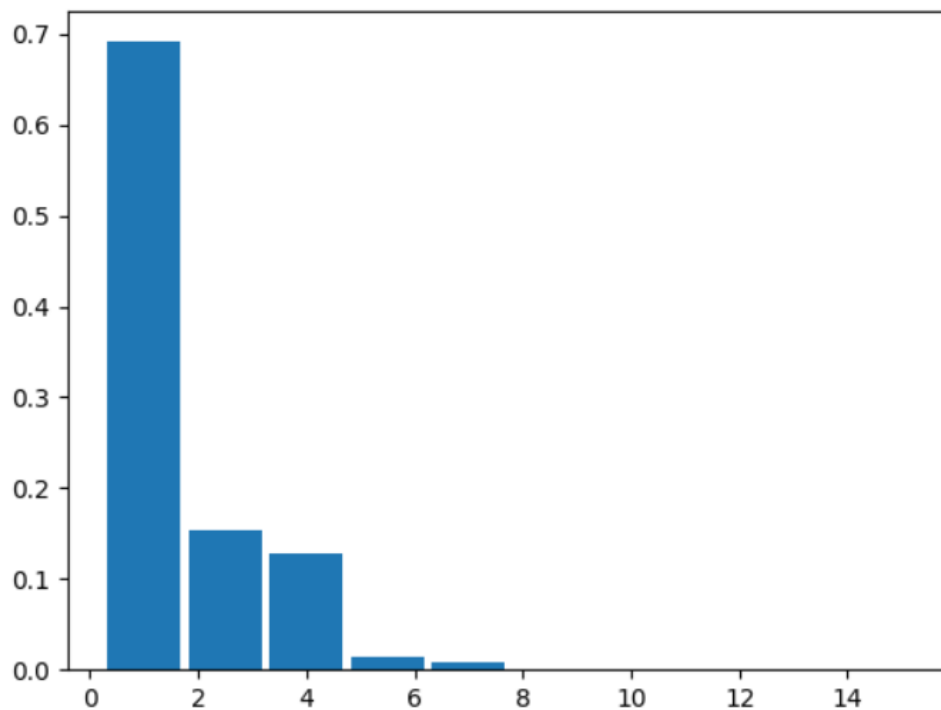
After the completion of the preprocessing data process for the distribution of numeric features representation, the numpy library command is enforced with its default settlements and separates the features values into ten bins. The general spectrum of the age range starts from age zero and reaches ages above eighty years old. Using the matplotlib library the histogram graph is depicted in Jupyter Notebook Figure 3. The distribution is normalized and it does not get like the gaussian one, as we can see the seven first bins present almost equal frequencies and an abrupt drop starts from the seventh bin that presents almost the half frequency of the sixth bin. The older people are rarer in the context in the particular dataset.

In Jupyter notebook figure 4 we have the normalized histogram of persons sequence number. The distribution differs from the distribution age and is constantly decreasing in terms of frequency as the bins correspond to higher attribute values and the first bin presents the highest difference by far in comparison to other bins' frequencies. Finally, the actual sequence persons number distribution is very diverged from the gaussian one and in this case the gaussian mechanism should be reevaluated if it represents the most effective one to create noise for a numeric feature with a corresponding distribution so as to lead to the best possible results of both privacy preservation and accuracy high standards.

For the graph's visualization the matplotlib library is used in order to present the normalized histograms that correspond to the density function of the data. The argument for the bars' width is set to 90% of the typical class width in order to present the exterior the most of the exterior edges. To be more specific, we use the `plt.bar()` function to create a bar chart. The first argument `bins_pn[:-1]` is a list of the edges of the histogram bins. The second argument `hist_pn` is a list of the counts for each bin. The third argument `width=(bins_pn[1]-bins_pn[0]) * 0.9` sets the width of each bar to 90% of the typical class width, which is calculated by subtracting the left bin edge from the right bin edge and multiplying by 0.9.



Jupyter Notebook Figure 1: Age normalized frequency histogram.



Jupyter Notebook Figure 2: Sequence Person number normalized frequency histogram.

### B3: Applying Differential Privacy to Numeric Columns with Gaussian Mechanism

In Exercise B, the same dataset that was anonymized in the previous step using the Amnesia tool is used. In this step, a random noise mechanism is applied to some of the numeric columns using the Gaussian mechanism in a way that preserves differential privacy. This step is crucial in protecting the privacy of individuals while still allowing for useful analysis of the dataset. The differential-privacy-library and its notebooks available on the IBM's GitHub are used for this purpose.

The Gaussian mechanism is imported from the `diffprivlib` library, which is a mechanism that adds random noise to the input data in a way that preserves differential privacy. The amount of noise added is determined by a privacy parameter called "epsilon" that specifies the maximum amount of privacy loss that we are willing to tolerate.

In this application, the Gaussian mechanism is applied to some of the numeric columns in the dataset in order to produce differentially private averages for each individual. The sensitivity of the mechanism and other key parameters, such as epsilon and delta, are also important in determining the degree of privacy loss and statistical accuracy of the query. The sensitivity of a query depends on the specific function being computed and the underlying data.

The code sets **epsilon to 1.0**, which means that a moderate degree of privacy loss is tolerated to achieve reasonable statistical accuracy. **Delta is set to 1e-5**, which is a very small value that ensures a very low probability of privacy loss due to random fluctuations. Finally, the **sensitivity of the mechanism is set to 5**, which is appropriate for the specific function that is computed.

After setting the parameters, an instance of the Gaussian mechanism is created from the `diffprivlib` library and its parameters are set to the values specified by the variables `epsilon`, `delta`, and `sen`.

Finally, two numeric vectors, `age` and `persons_num`, are created using the data dataframe, which contains the values from the `Age` column and the `Person Sequence Number` column of the data dataframe, respectively. The `randomise` function of the Gaussian mechanism from the `diffprivlib` library is then applied to the `age` and `persons_num` numeric vectors, which adds random noise to each value in the vector while preserving differential privacy. The resulting values after applying the randomised function to `age` are stored in the `vals_age_gauss` vector, and the resulting values after applying the randomised function to `persons_num` are stored in the `vals_pn_gauss` vector.



## ***B4: Calculating Differentially Private Averages for Numeric Features in a Dataset***

In this step of the notebook, the focus is on calculating differentially private averages for the numeric features in the dataset. The previously applied k-anonymity and random noise addition techniques were aimed at preserving the privacy of individuals in the dataset. Differential privacy is a privacy guarantee that ensures that the output of a query on a dataset does not reveal any sensitive information about individuals.

The differential privacy library provided by IBM is used to calculate differentially private averages, and the results are compared with the original averages. The average age of all individuals in the dataset after the implementation of Gaussian random noise is found to be 38.34, and the average sequence persons number is 2. In comparison, the average age of all individuals in the initial dataset is 38.27, and the average sequence persons number is 2.

From the results, it can be interpreted that the addition of Gaussian random noise has slightly affected the average age of individuals in the dataset, as the differentially private average is slightly higher than the original average. However, the difference between the two averages is relatively small. Additionally, the average sequence persons number remained the same after the addition of noise. This shows that the differentially private averages calculated using the noisy data can provide a reasonable approximation of the true averages in the dataset while still preserving privacy.

## ***B5: Analyzing the Effect of Differential Privacy on Numeric Features Distribution***

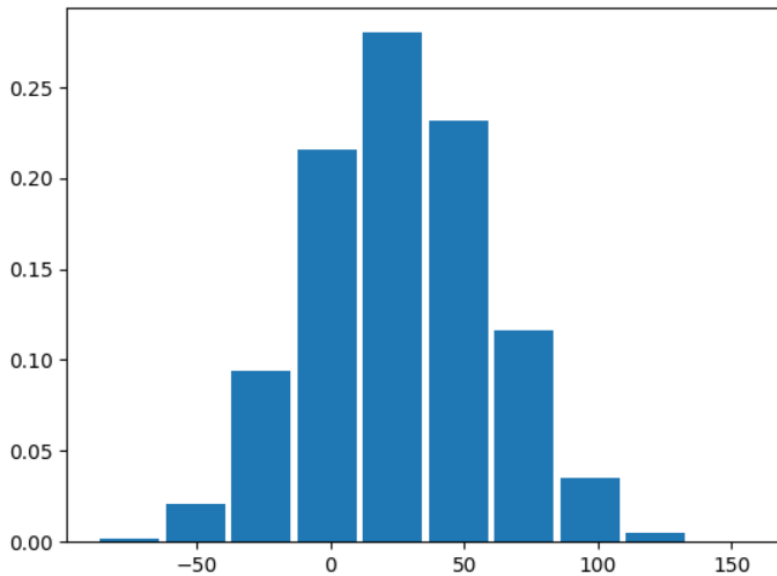
In the final step of Exercise B, we will plot the distribution of the numeric features after adding random noise using the Gaussian mechanism with different values of the privacy parameter epsilon ( $\epsilon$ ). We will also comment on the effect of differential privacy on the results. This step will help us understand how the addition of noise affects the distribution of data and how it impacts the privacy guarantees provided by differential privacy. We will be using the differential privacy library provided by IBM for this step, which offers a range of functions for performing differentially private analysis.

The gaussian mechanism is selected for creating the added noise in the numeric feature with the  $\epsilon$  parameter set in the highest edge of the spectrum defined by the `diffprivlib` library that sets this spectrum to  $[0,1]$ . When using the Gaussian mechanism in differential privacy, the added noise follows a Gaussian distribution. Therefore, the histogram of the differentially private values will have a bell shape that is like a Gaussian distribution. In other words, adding Gaussian noise to data is equivalent to convolving the original data with a Gaussian kernel, which results in a smoothed version of the original data. The degree of smoothing is controlled by the standard deviation of the Gaussian distribution, which is a parameter of the mechanism. The amount of noise added to each data point is proportional to the sensitivity of the function being computed (e.g., mean, sum, count) and inversely proportional to the privacy budget. A larger privacy budget results in less noise being added and therefore a smaller deviation from the original data, but with lower privacy guarantees.

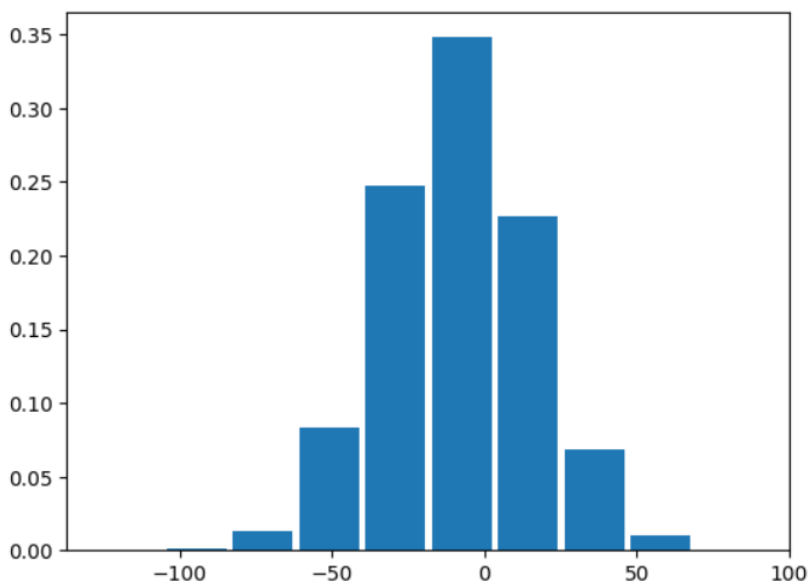
The first  $\epsilon$  value used is 1 and the histograms for both numeric features are created using the aforementioned hyperparameter tuning. In general, a smaller  $\epsilon$  value provides stronger privacy guarantees but requires more noise to be added to the data, resulting in lower accuracy. Conversely, a larger  $\epsilon$  value results in less noise being added but provides weaker privacy guarantees. For the Gaussian mechanism in `diffprivlib`, the amount of noise added is proportional to  $1/\epsilon$ . Therefore, as  $\epsilon$  decreases, the amount of noise added increases and the privacy guarantees become stronger. It is important to note that the choice of  $\epsilon$  depends on the specific application and the desired trade-off between privacy and accuracy. A smaller  $\epsilon$  value may be appropriate for applications that require strong privacy protection, while a larger  $\epsilon$  value may be acceptable for applications that require higher accuracy but can tolerate lower levels of privacy protection.

The histogram shapes of the differentiated features present the gaussian shape as expected and Jupyter Notebook Figures 4 and 5 present this. The histograms correspond to the less possible noise enforced to the numeric data using the gaussian mechanism however, the differentiation of the histograms format is obvious and substantial. Consequently, the data values are differentiated in a serious extent. Another important aspect that should be taken into consideration is that the noise added results to the

appearance of negative values in both numeric features as the random mechanism gives negative values although the features' semantics do not allow negative values. In another comparative case the random mechanism should be parametrize in a way that negative noise is not allowed to see if this result to less differentiated values. Another reason for which this topic is important concerns the analysis queries that could use differentiated data and deliver negative results and lead to false analysis result.



Jupyter Notebook Figure 3: Age normalized frequency histogram after the added noise is implemented to the feature.



Jupyter Notebook Figure 4: Sequence Personal Number normalized frequency histogram after the added noise is implemented to the feature.

The next steps include the calculation of the histogram error for both numeric features cases. The histogram error corresponds to error between the original histogram and the differentially private histogram. If this error is high, this means that the differentially private mechanism has added a significant amount of noise to the data, which can result in lower accuracy. However, the added noise also provides stronger privacy guarantees. This is because the amount of noise added is inversely proportional to the privacy budget or epsilon value, which determines the strength of privacy

guarantees. A smaller epsilon value results in stronger privacy guarantees but more noise added to the data, which can lead to higher error rates.

In general, if the error is low, then the accuracy of the differentially private histogram is likely to be acceptable for many applications. However, if the error is high, then the accuracy may be too low for many applications. Therefore, it is important to carefully consider the trade-off between privacy and accuracy when selecting the epsilon value and evaluating the accuracy of differentially private histograms. More specifically, high error may indicate that the privacy budget or epsilon value was set too low, resulting in too much noise being added to the data. It is important to balance privacy guarantees with the desired level of accuracy for the given task. If the error is unacceptably high, one option may be to increase the privacy budget or epsilon value to reduce the amount of noise added and improve accuracy. This approach should be done with caution and after carefully considering the desired level of privacy and the potential risks of data exposure.

In our case the histogram error is calculated to 0.69 for the age attribute and to 1.91 for the sequence personal number attribute. The smaller range of the sequence personal number feature is a justification factor for the highest histogram errors appearing. In both cases the error is significant but we cannot label it as acceptable or not acceptable as the overall context and goals of a potential statistical analysis carried out in the dataset will lead to the estimation and settlement of accepted error ranges for each numeric a feature and the specific use case and the consequences of inaccurate data. In some cases, even a small amount of error may be unacceptable. In other cases, a higher error rate may be tolerable if the privacy guarantees provided by differential privacy are more important.

The hyper tuning of the differential privacy mechanism is a process of repetitive steps that has the main purpose of optimizing the method applied in particular dataset and the trade-off between the privacy loss and the accuracy gained in statistical queries. In the context of our hyper tuning process different values of  $\epsilon$  were tested. For all distinct values of the  $\epsilon$  parameter a new corresponding mechanism were created and fitted to the numeric features causing different amount of noise to be added. Then, the new bins edges for the histograms after the noise is added are calculated along with the frequency of each separate bin. On the final step the histogram error between the noise data and the initial data is calculated for each different case as the measure of evaluation of the method's impact in the data. Five  $\epsilon$  values are used and tested in order to present the different parameter's effects. In particular,  $\epsilon$  values tested are 0.01, 0.1, 0.5, 0.8, 1. As the following figure 6 shows the different histogram error values for the different  $\epsilon$  values.

```
Total histogram error for the age attribute: 0.825964 for  $\epsilon = 0.01$ 
Total histogram error for the sequence person number attribute: 1.917530 for  $\epsilon = 0.01$ 
Total histogram error for the age attribute: 0.828055 for  $\epsilon = 0.1$ 
Total histogram error for the sequence person number attribute: 1.917842 for  $\epsilon = 0.1$ 
Total histogram error for the age attribute: 0.788933 for  $\epsilon = 0.5$ 
Total histogram error for the sequence person number attribute: 1.918220 for  $\epsilon = 0.5$ 
Total histogram error for the age attribute: 0.737289 for  $\epsilon = 0.8$ 
Total histogram error for the sequence person number attribute: 1.918353 for  $\epsilon = 0.8$ 
Total histogram error for the age attribute: 0.697411 for  $\epsilon = 1$ 
Total histogram error for the sequence person number attribute: 1.918309 for  $\epsilon = 1$ 
```

*Jupyter Notebook Figure 5: Histogram errors results for different epsilon parameters values.*

As the results present the histogram error augments for both attributes as the  $\epsilon$  parameter gets smaller. This is expected as this parameter determines the privacy loss level of the method. The smaller the epsilon the bigger the noise added and consequently the smaller the privacy loss as the more privacy guarantee is attained. On the other hand, as the  $\epsilon$  gets higher less noise is added to the data and higher privacy loss is presented in exchange of higher accuracy as well. For  $\epsilon = 0.01$  the smaller tested value age has a histogram error of 0.82 and sequence person number has an error of 1.91. Finally, age has an error of 0.69 quite smaller when the  $\epsilon = 1$  while, sequence person number has the smallest error attained through the aforementioned trials but in the case of this more restricted numeric feature the error reduction is much more restrained.



## Deliverables Archive

In this section, we provide a brief overview of the deliverables archive that contains all the outputs generated throughout the analysis process. To be more precise, in the deliverable's ZIP file '**Data Anonymization Exercise.zip**' you can find the following:

- The current PDF file, '**Business and Privacy Issues in Data Analysis.pdf**'
- The Jupyter Notebook file containing the Exercise B code for the Differential Privacy questions 2-5, '**Differential\_Privacy.ipynb**'
- The above executable of Jupyter Notebook is also saved as an HTML readable file, in order to have an exact copy of the notebook in case a library or Python version has been updated or deprecated, '**Differential Privacy.html**' and a printed PDF file, '**Differential\_Privacy - Jupyter Notebook.pdf**'
- The programming language R file that was used prior to Exercise A for the data preprocessing stage, '**pubs.R**'
- The '**Datasets**' folder that contains various significant databases that were used during this assignment.
- The '**Given Data**' folder that contains the original given data.
- The '**Hierarchies**' folder that contains all the hierarchies created and used for this assignment as well as the basic ones.
- The '**Amnesia Process Pictures**' that contains all the screenshots and figures used in the current documentation as well as more figures that cut off to keep the documentation simple.

## References

The following references were used throughout the analysis process to provide theoretical and practical guidance and the tools used in this report.

1. Dwork, C. (2008). Differential privacy: A survey of results. In Theory and Applications of Models of Computation (pp. 1-19). Springer.
2. "A tutorial on re-identification attacks" by El Emam et al. (2011) - <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3113384/>
3. "Re-identification of individuals in genomic data-sharing beacons via allele inference" by Erlich and Narayanan (2014) - <https://www.nature.com/articles/ng.3052>
4. "Re-identification of home addresses from spatial locations anonymized by Gaussian skew" by Yang et al. (2019) - <https://arxiv.org/abs/1912.10116>
5. "Differential Privacy: A Survey of Results" by Dwork (2008) - <https://www.cs.cmu.edu/~dwork/DP-survey.pdf>
6. "The Algorithmic Foundations of Differential Privacy" by Dwork and Roth (2014) - <https://www.cis.upenn.edu/~aaroht/Papers/privacybook.pdf>
7. "On the trade-off between privacy and utility in data publishing" by Machanavajjhala et al. (2008) - <https://dl.acm.org/doi/10.1145/1376616.1376626>
8. "On the Complexity of Differential Privacy" by Goyal, Kamath, and Thakurta (2018) - <https://arxiv.org/pdf/1803.01461.pdf>
9. "Deep Learning with Differential Privacy" by Abadi et al. (2016) - <https://arxiv.org/pdf/1607.00133.pdf>
10. IBM. (n.d.). Differential Privacy Library. [GitHub Repository]. Retrieved from <https://github.com/IBM/differential-privacy-library>
11. IBM. (n.d.). Differential Privacy Library: Notebooks. [GitHub Repository]. Retrieved from <https://github.com/IBM/differential-privacy-library/tree/main/notebooks>
12. Amnesia Anonymization Tool. (n.d.). OpenAIRE. [online] Retrieved from <https://amnesia.openaire.eu/>
13. R Core Team. (2023). R: A language and environment for statistical computing. R Foundation for Statistical Computing. Retrieved from <https://www.R-project.org/>
14. RStudio Team (2023). RStudio: Integrated Development Environment for R. RStudio, PBC, Boston, MA URL <http://www.rstudio.com/>
15. Van Rossum, G., Warsaw, B., Coghlan, N., & Beazley, D. M. (2021). The Python Language Reference, Version 3.10. Python Software Foundation. Retrieved from <https://www.python.org/downloads/>
16. Project Jupyter. (2023). Jupyter Notebook. Retrieved from <https://jupyter.org/>
17. Walt, S. van der, Colbert, S. C., & Varoquaux, G. (2011). The NumPy array: a structure for efficient numerical computation. Computing in Science & Engineering, 13(2), 22–30. <https://doi.org/10.1109/MCSE.2011.37>
18. Harris, C. R. et al. (2020) 'Array programming with NumPy', Nature, 585(7825), pp. 357-362. Available at: <https://doi.org/10.1038/s41586-020-2649-2>
19. Erlingsson, Ú. and Tran, T. (2019) 'The Gradient Perturbation Mechanism: Differential Privacy for Gradient Descent', in H. Wallach et al. (eds) Advances in Neural Information Processing Systems, 32, pp. 7514-7524.
20. Hunter, J. D. (2007) 'Matplotlib: A 2D graphics environment', Computing in Science & Engineering, 9(3), pp. 90-95. Available at: <https://doi.org/10.1109/MCSE.2007.55>
21. NumPy 'NumPy Documentation', Available at: <https://numpy.org/doc/stable/>



- 
22. Pandas Development Team 'pandas documentation', Available at:  
<https://pandas.pydata.org/docs/>
  23. Diffprivlib Developers 'Diffprivlib Documentation', Available at:  
<https://diffprivlib.readthedocs.io/en/latest/>
  24. Matplotlib Development Team 'Matplotlib Documentation', Available at:  
<https://matplotlib.org/stable/contents.html>
  25. Vrije Universiteit Amsterdam (2018) VU Data Conversations | Manolis Terrovitis | Amnesia – Data Anonymisation Made Easy. Available at:  
[https://www.youtube.com/watch?v=6wJKDh1EBLA&ab\\_channel=VrijeUniversiteitAmsterdam](https://www.youtube.com/watch?v=6wJKDh1EBLA&ab_channel=VrijeUniversiteitAmsterdam)
  26. OpenAIRE (2020) OpenAIRE webinar | Manolis Terrovitis : Amnesia. Available at:  
[https://www.youtube.com/watch?v=0lo6c1MPOY&ab\\_channel=OpenAIRE\\_eu](https://www.youtube.com/watch?v=0lo6c1MPOY&ab_channel=OpenAIRE_eu)
  27. Terrovitis, M. (2023) OpenAIRE Webinar - AMNESIA: High-accuracy Data Anonymization. Available at: [https://www.youtube.com/watch?v=pgtLY1r9eeM&ab\\_channel=OpenAIRE\\_eu](https://www.youtube.com/watch?v=pgtLY1r9eeM&ab_channel=OpenAIRE_eu)
  28. An Nguyen: Understanding Differential Privacy. From Intuitions behind a Theory to a Private AI Application. Available at: [Understanding Differential Privacy | by An Nguyen | Towards Data Science](#)